

Universidade de Brasília – UnB

Engenharia Elétrica

# **Controladores de Carga MPPT para Painéis Solares**

**Autor: Lucas Vasconcelos de Moraes**

**Orientador: Prof. Daniel Chaves Café**

**Brasília, DF**

**2018**



Lucas Vasconcelos de Moraes

## **Controladores de Carga MPPT para Painéis Solares**

Trabalho de conclusão de curso submetido ao departamento de Engenharia Elétrica da Universidade de Brasília, como requisito parcial para obtenção do Título de Engenheiro Eletricista.

Universidade de Brasília – UnB

Orientador: Prof. Daniel Chaves Café

Brasília, DF

2018

---

Lucas Vasconcelos de Moraes

Controladores de Carga MPPT para Painéis Solares / Lucas Vasconcelos de  
Moraes. – Brasília, DF, 2018-

82 p. : 30 cm.

Orientador: Prof. Daniel Chaves Café

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB  
, 2018.

1. MPPT. 2. Controladores de Carga. I. Prof. Daniel Chaves Café. II.  
Universidade de Brasília. III. Controladores de Carga MPPT para Painéis Solares

CDU 02:141:005.6

---

Lucas Vasconcelos de Moraes

## **Controladores de Carga MPPT para Painéis Solares**

Trabalho de conclusão de curso submetido ao departamento de Engenharia Elétrica da Universidade de Brasília, como requisito parcial para obtenção do Título de Engenheiro Eletricista.

Trabalho aprovado. Brasília, DF, 02 de Julho de 2018:

---

**Prof. Daniel Chaves Café**  
Orientador

---

**Prof. Rafael Amaral Shayani**  
Examinador Interno

---

**Prof. José Edil Guimarães de  
Medeiros**  
Examinador Interno

---

**Paulo Vinícius Moura Coelho**  
Examinador Externo

Brasília, DF  
2018



# Resumo

A demanda por energias renováveis vem crescendo no atual contexto mundial e faz-se cada vez mais importante o estudo sobre novas tecnologias com o objetivo de obter-se uma energia mais limpa e eficiente. A energia solar é definitivamente um excelente caminho para obtenção de energia dita sustentável e é provinda de uma grande fonte energética na natureza. Em geral, existem duas maneiras de alocar-se a potência advinda de painéis solares. A primeira é direcionar o excedente de energia obtida para a rede tradicional do sistema elétrico. Já a segunda trata do armazenamento da energia em baterias. Esse trabalho de conclusão de curso tem por objetivo o estudo dos sistemas de painéis solares alocados fora da rede tradicional de energia e o desenvolvimento de um controlador de carga com algoritmo de rastreamento do ponto de máxima transferência de potência. A maximização da potência de painéis solares passa pelo uso de ferramentas fundamentais como conversores de tensão DC/DC, sensores e algoritmos de controle de carga. O rastreamento do ponto de máxima potência é feito com circuitos sensores de corrente e tensão no painel solar, sendo o controle feito por um microcontrolador MSP430.

**Palavras-chaves:** Controle de Carga. Painéis Solares. Eficiência Energetica. Conversão DC/DC. Energias Renováveis.





# Abstract

The demand of renewable energy has been increasing in a response to the market's demands on sustainable energy generation. The solar panel is definitely a great way to obtain power from an abundant source of energy in nature. Generally, there are two ways of addressing the power from photovoltaic cells: In-grid and off-grid systems. This work proposes a study about off-grid systems and the development of a charge controller using the MPPT (Maximum Power Point Tracking) algorithm. The tracking of the maximum power point is done with the help of a voltage and current measurement circuit and it is driven by a MSP430 microcontroller.

**Key-words:** Panel Solar, MPPT, Renewable Energy, Pv cells.



# Lista de ilustrações

Figura 1 – Configuração de Sistema Solar. FONTE:(SAHA, 2017)	16
Figura 2 – Configuração Sunblazer. FONTE:(SAHA, 2017)	16
Figura 3 – AC Microgrid. FONTE:(SAHA, 2017)	17
Figura 4 – DC Microgrid. FONTE:(SAHA, 2017)	17
Figura 5 – Esquemático de Rede solar Off-Grid via MPPT	19
Figura 6 – Sistema equivalente de uma célula fotovoltaica com modelo exponencial	21
Figura 7 – Modelo de painel solar e bateria simplificados	22
Figura 8 – Simulação (Tensão bateria Vs Corrente de entrada)	23
Figura 9 – Simulação (Tensão bateria Vs Potência Entregue a Bateria)	23
Figura 10 – Tensão bateria Vs Corrente Painel Solar. FONTE: (B.V, 2014)	24
Figura 11 – Tensão Sobre Carga Resistiva Vs Corrente Painel Solar. FONTE: (B.V, 2014)	24
Figura 12 – Conversor <i>BUCK</i> (abaixador) FONTE:(HARTS, 2011)	25
Figura 13 – Forma de onda conversor <i>BUCK</i> (abaixador) FONTE:(HARTS, 2011)	27
Figura 14 – Forma de onda conversor <i>boost</i> (elevador) FONTE:(HARTS, 2011)	29
Figura 15 – Forma de onda conversor buck-boost FONTE:(HARTS, 2011)	30
Figura 16 – Forma de onda conversor <i>SEPIC</i> FONTE:(HARTS, 2011)	32
Figura 17 – Modelo MSP430 Texas Instrumentos. FONTE:(TI, 2008)	33
Figura 18 – PWM em controle de tensão. FONTE:(HARTS, 2011)	34
Figura 19 – Esquemático do timer A MSP430F5529.FONTE:(TI, 2008)	36
Figura 20 – Algoritmo Perturba e Observa. FONTE:(YADAV1; S.THIRUMALIAH2; G.HARITHA3, 2012)	37
Figura 21 – Algoritmo Condutância Incremental.FONTE:(YADAV1; S.THIRUMALIAH2; G.HARITHA3, 2012)	38
Figura 22 – Painel solar 12V 22W	39
Figura 23 – Painel solar 5V 1W	40
Figura 24 – Gráfico V x I obtido em laboratório	46
Figura 25 – Topologia SEPIC	47
Figura 26 – Simulação f=100kHz D=0.2	47
Figura 27 – Simulação Abaixador de Tensão f=100kHz D=0.4 abaixador	47
Figura 28 – Simulação Abaixador de Tensão f=100kHz D=0.5	48
Figura 29 – Simulação Elevador de Tensão f=100kHz D=0.8	48
Figura 30 – Montagem Realizada em Laboratório Lab 1 - SG11	49
Figura 31 – Relação Tensão na Saída do painel vs onda PWM para D=38,2%	49
Figura 32 – <i>Duty Cycle</i> Vs Tensão de saída no painel	50
Figura 33 – Gráfico <i>Duty Cycle</i> Vs Corrente na saída do painel	51

Figura 34 – Tensão Vs Corrente na saída do painel . . . . .	51
Figura 35 – Gráfico <i>Duty Cycle</i> Vs Potência na saída do painel solar . . . . .	52
Figura 36 – Sensor de Corrente . . . . .	53
Figura 37 – Circuito de Sensor de Corrente uGold . . . . .	53
Figura 38 – Circuito Sensor de Corrente Implementado . . . . .	54
Figura 39 – Circuito Mudança de Referência . . . . .	55
Figura 40 – Simulação Tensão Painel vs (Tensão na Saída do Sensor)/(Corrente em $R_{sens}$ ) . . . . .	55
Figura 41 – Onda PWM com $D=15.8\%$ e Tensão na Saída do Sensor de Corrente . . . . .	56
Figura 42 – Onda PWM com $D=22.4\%$ e Tensão na Saída do Sensor de Corrente . . . . .	56
Figura 43 – Onda PWM com $D=45.4\%$ e Tensão na Saída do Sensor de Corrente . . . . .	57
Figura 44 – Relação entre <i>duty cycle</i> vs (Tensão no sensor)/(Corrente na saída do painel) . . . . .	57
Figura 45 – Divisor Resistivo . . . . .	58
Figura 46 – Conversão Tensão [Volts] x Amostras Digitais [N] . . . . .	59
Figura 47 – Conversão Corrente [mA] x Amostras Digitais [N] . . . . .	60
Figura 48 – Conversão Potência [mW] x Amostras Digitais [N] . . . . .	60
Figura 49 – Sinal Digitalizado Sensor de Corrente Funcionando com o Painel Solar (Ni x tempo) . . . . .	61
Figura 50 – Sinal Digitalizado Sensor de Tensão Funcionando com o Painel Solar (Nv x tempo) . . . . .	61
Figura 51 – Relação do Sinal de Tensão Variando de Acordo com Algoritmo P&O com luz contante (Nv x tempo) . . . . .	62
Figura 52 – Relação do Sinal de Corrente Variando de Acordo com Algoritmo P&O com luz constante (Ni x tempo) . . . . .	62
Figura 53 – Relação do Sinal de Potência Variando de Acordo com Algoritmo P&O com luz constante (Np x tempo) . . . . .	63
Figura 54 – Relação do Sinal de Tensão Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa (Nv x tempo) . . . . .	63
Figura 55 – Relação do Sinal de Corrente Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa (Ni x tempo) . . . . .	63
Figura 56 – Relação do Sinal de Potência Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa (Np x tempo) . . . . .	63

# Lista de tabelas

Tabela 1 – Parâmetros MSP430F5529 . . . . .	34
Tabela 2 – Parâmetros Painel Solar 12V 22W . . . . .	40
Tabela 3 – Parâmetros Mini Painel Solar 5V 1W . . . . .	40
Tabela 4 – Relação de Componentes Utilizados . . . . .	41
Tabela 5 – Relação de Componentes Utilizados . . . . .	42
Tabela 6 – Relação entre a tensão aplicada e corrente no painel solar . . . . .	45
Tabela 7 – Relação PWM Vs Potência Extraída do Painel Solar . . . . .	50
Tabela 8 – Relação de Tensão do Sensor com Corrente Extraída de Painel Solar . . . . .	57



# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>15</b>
<b>1.1</b>	<b>Contextualização</b>	<b>15</b>
<b>1.2</b>	<b>Objetivos Gerais</b>	<b>19</b>
<b>1.3</b>	<b>Objetivos Específicos</b>	<b>20</b>
1.3.1	Implementação conversor DC/DC	20
1.3.2	Sensor de Corrente e Tensão	20
1.3.3	Algoritmo de Máxima Transferência de Potência	20
1.3.4	Teste de Protótipo Solar via MPPT	20
<b>2</b>	<b>EMBASAMENTO TEÓRICO</b>	<b>21</b>
<b>2.1</b>	<b>Painéis Solares</b>	<b>21</b>
2.1.1	Modelo para Painéis Fotovoltaicos	21
2.1.2	Simulações do Modelo Fotovoltaico	22
2.1.3	Ponto Máximo de Potência	23
<b>2.2</b>	<b>Conversores DC-DC</b>	<b>25</b>
2.2.1	Conversor <i>Buck</i>	25
2.2.2	Conversor <i>Boost</i>	28
2.2.3	Conversor Buck-Boost	30
2.2.4	Conversor com Indutância Simples no Primário ( <i>SEPIC</i> )	31
<b>2.3</b>	<b>Microcontroladores</b>	<b>33</b>
2.3.1	MSP430	33
2.3.2	Modulação por Largura de Pulso	34
2.3.3	Temporizadores	35
<b>2.4</b>	<b>Algoritmo de Máxima Transferência de Potência</b>	<b>36</b>
2.4.1	Perturba e Observa	36
2.4.2	Condutância Incremental	37
<b>3</b>	<b>METODOLOGIA</b>	<b>39</b>
<b>3.1</b>	<b>Metodologia de Medidas</b>	<b>39</b>
3.1.1	Caracterização de Painel Solar	39
3.1.2	Conversor/Painel Solar	41
3.1.3	Sensores	42
3.1.4	Conversão Analógica Digital e Algoritmo P&O	42
<b>3.2</b>	<b>Ferramentas Utilizadas</b>	<b>43</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>45</b>

<b>4.1</b>	<b>Painel Solar</b>	<b>45</b>
<b>4.2</b>	<b>Sistema Conversor/Painel Solar</b>	<b>46</b>
4.2.1	Simulações do Circuito DC/DC	47
4.2.2	Montagem do Sistema DC/DC - Painel - Baterias	49
<b>4.3</b>	<b>Sensores</b>	<b>52</b>
4.3.1	Simulações Sensor de Corrente	52
4.3.2	Montagem Sensor de Corrente	56
4.3.3	Sensor de Tensão	58
<b>4.4</b>	<b>Microcontrolador</b>	<b>59</b>
4.4.1	Conversão Analógico Digital	59
4.4.2	Algoritmo Perturba e Observa	62
<b>5</b>	<b>CONCLUSÃO</b>	<b>65</b>
	<b>REFERÊNCIAS</b>	<b>67</b>
	<b>APÊNDICE A – APÊNDICE</b>	<b>69</b>
<b>A.1</b>	<b>Código Variação da Taxa de Trabalho via Botões: MSP430</b>	<b>69</b>
<b>A.2</b>	<b>Código Conversão ADC via Botões: MSP430</b>	<b>72</b>
<b>A.3</b>	<b>Código Algoritmo Perturba e Observa</b>	<b>77</b>



# 1 Introdução

## 1.1 Contextualização

Dados da Agência Internacional de Energia ([IEA, 2017](#)) divulgados em 2015 estimam que cerca de 1.1 bilhões de pessoas vivem sem energia elétrica em todo o mundo. Diante desse elevado número de pessoas vivendo à margem da economia mundial, extremamente dependente de fontes de energia elétrica, faz-se necessário encontrar alternativas viáveis para suprir a demanda atual de energia. A energia provinda do sol e captada através de painéis solares, mostra-se como boa alternativa às fontes não renováveis e apresenta alto potencial de distribuição a lugares remotos.

Existem diversas abordagens possíveis em relação aos painéis solares. Sistemas conectados ao sistema elétrico tradicional demonstram ser uma alternativa sustentável em se tratando de fontes renováveis de energia elétrica, porém não são aplicáveis em lugares afastados, nos quais o sistema tradicional de potência não chega. Frente a tal obstáculo surge o sistema fora da rede, mais conhecidos como *off-grid*, no qual baterias são usadas para armazenar a potência extraída da radiação solar via painéis fotovoltaicos.

O IEEE realiza um programa denominado *Smart Village* no qual busca-se levar energia para zonas rurais por meio de painéis solares de energia. Na presente data existem projetos já finalizados pela *Smart Village* ([SAHA, 2017](#)). Um dos projetos, denominado *sunblazer* tem como objetivo carregar baterias portáteis para que moradores de zonas rurais possuam uma fonte de energia para pequenas tarefas do dia a dia.

A implementação do sistema *off-grid* pode ser feito de diferentes formas. As principais implementações feitas via programa *Smart Villages*, hoje em funcionamento, são:

- Configuração *Solar Home*

Painéis solares carregam baterias que através de controladores de carga, com ou sem a tecnologia MPPT<sup>1</sup>, são capazes de abastecer parte da energia elétrica consumida em uma casa.

---

<sup>1</sup> Maximum Power Point Tracker

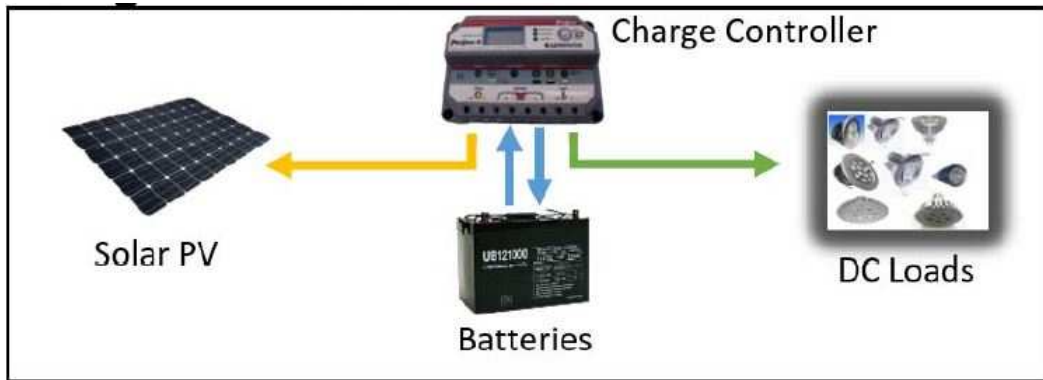


Figura 1 – Configuração de Sistema Solar. FONTE:(SAHA, 2017)

- Configuração *Sunblazer*

Painel solar ligado a um controlador de carga para abastecer baterias de 24V. As baterias são em seguida submetidas a conversores DC/DC e podem carregar baterias removíveis para a população local.



Figura 2 – Configuração Sunblazer. FONTE:(SAHA, 2017)

- Configuração *AC Microgrid*

Configuração capaz de sincronizar um gerador com painéis solares por meio de um sincronizador inversor que é submetido a um adaptador AC/DC para então carregar baterias para uso geral.

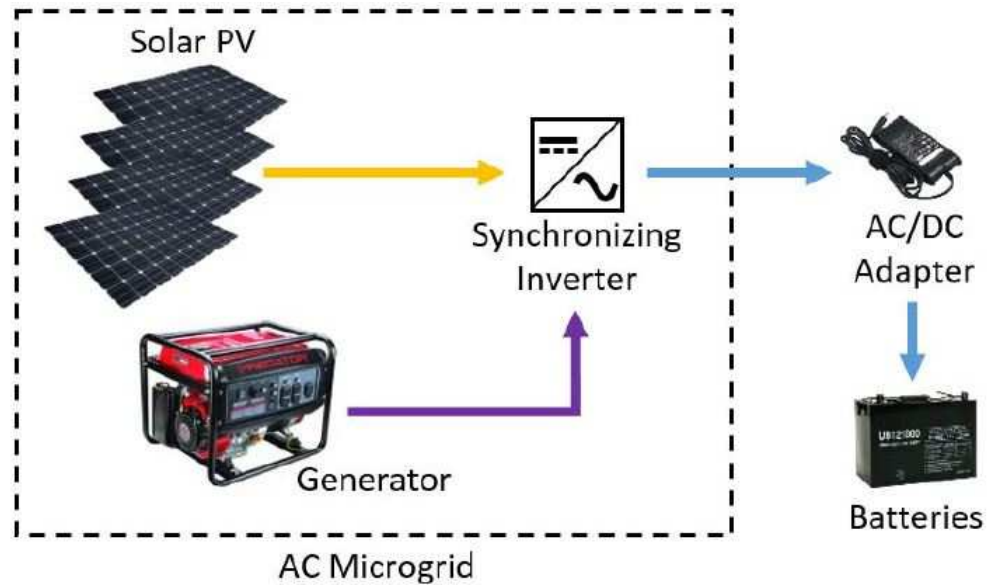


Figura 3 – AC Microgrid. FONTE:([SAHA, 2017](#))

- Configuração *DC Microgrid*

Painel solar ligado a conversor DC/DC para posteriormente ser submetido a um controlador de carga que pode abastecer baterias para uso geral.

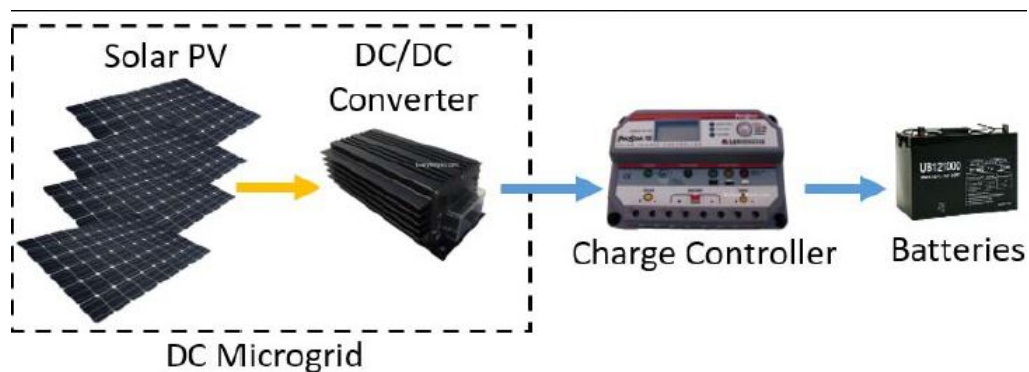


Figura 4 – DC Microgrid. FONTE:([SAHA, 2017](#))

Pode-se observar em todas essas configurações o uso de carregadores e conversores DC/DC. A partir das configurações acima, nota-se um grande limitador em criar-se um sistema universal para a integração de diversos tipos de sistemas *off-grid*: a não existência de um controlador de carga universal que possa suprir todas as diferentes necessidades de cada tipo de configuração.

Dado esse fator limitante, a proposta do seguinte trabalho relaciona-se ao estudo do funcionamento básico de controladores de carga inteligentes, funcionando com MPPT e também de conversores DC-DC. Controladores universais de carga são fundamentais para o ajuste de parâmetros importantes como corrente, tensão e da potência gerada

pelos painéis solares. A extrema importância desses equipamentos na procura pela máxima transferência de potência gerou a motivação necessária para que um estudo mais aprofundado sobre esses componentes fosse realizado.

O objetivo do trabalho é a elaboração de um dispositivo controlador de carga via algoritmo de máxima transferência de potência. Serão aprofundados aspectos teóricos e práticos para a implementação de um sistema fotovoltaico controlado visando o máximo aproveitamento de potência. É evidente que em se tratando de tema tão amplo e complexo não haverá tempo hábil para o estudo e a implementação de um controlador de carga universal, cuja importância é nítida no atual contexto mundial. Entretanto, será introduzido o funcionamento teórico e prático de um sistema fotovoltaico microcontrolado. Primeiramente será embasado teoricamente o funcionamento e comportamento de painéis solares. Após a análise da problemática relativa ao ponto máximo de potência, será apresentado o embasamento teórico de conversores DC/DC e o algoritmo a ser usado na busca por máxima eficiência de potência. Além disso, será necessário o uso de microcontrolador para controle de parâmetros como corrente, tensão e taxa de trabalho<sup>2</sup> do sistema.

Segue a lista de tópicos que buscaremos introduzir ao longo do trabalho:

- Comportamento de Painéis Solares
- Conversores DC/DC
- Carregadores de PV <sup>3</sup>
- Algoritmo MPPT
- Microcontroladores
- Controle de conversores DC/DC via PWM <sup>4</sup>
- Possível implementação de um carregador inteligente controlado via MPPT

---

<sup>2</sup> Tradução para duty cycle

<sup>3</sup> Painel Fotovoltaico

<sup>4</sup> Pulse-Width Modulation

## 1.2 Objetivos Gerais

Será implementado um controlador de carga via algoritmo inteligente de maximização de potência. A implementação do painel solar resultará na concretização dos conhecimentos adquiridos ao longo do projeto para exemplificar o método de máxima transferência de potência.

Ao final do projeto vamos realizar a implementação do seguinte esquemático:

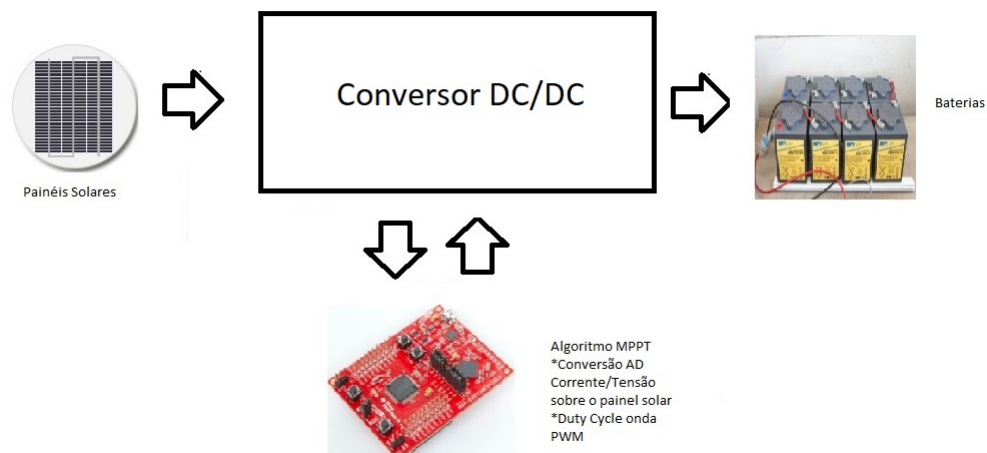


Figura 5 – Esquemático de Rede solar Off-Grid via MPPT

O esquemático na Figura 5 mostra de forma simplificada o processo que será construído. O sistema consiste em painéis solares de energia que absorvem radiação solar e geram potência para carregar as baterias de um sistema de captação fora da rede. Para forçar os painéis de energia a funcionarem no seu ponto máximo de potência será utilizado um conversor DC/DC controlado via PWM por um microcontrolador. O microcontrolador será responsável pelo controle do *duty cycle* dos conversores e também por analisar dados colhidos, via sensores de corrente e tensão, do painel solar.

## 1.3 Objetivos Específicos

Consistem em objetivos específicos:

### 1.3.1 Implementação conversor DC/DC

Primeiramente, faz-se necessária a implementação de um conversor DC/DC, que será responsável pelo ajuste da tensão de saída do painel solar para a tensão de máxima transferência de potência. Tal etapa trata-se de um passo fundamental ao sucesso de todo o projeto, uma vez que é necessário garantir que o conversor funcione com razoável margem de segurança, bem como com elevada eficiência. Nessa etapa, os parâmetros teóricos serão ajustados para a construção de um conversor cujo comportamento será observado em laboratório em diferentes situações. Após a observação de diferentes topologias clássicas de conversores DC/DC, será escolhida a opção que melhor se encaixe no objetivo do trabalho, julgando quesitos como eficiência energética, complexidade de implementação, custos, aplicabilidade e todo critério julgado pertinente futuramente.

### 1.3.2 Sensor de Corrente e Tensão

Nesse passo do projeto, busca-se implementar um sensor de corrente e tensão capaz de quantificar valores referentes a saída de um painel solar. Nessa etapa será necessário estudar uma forma razoável de implementar-se tais sensores tendo em vista os equipamentos necessários, bem como os empecilhos observados ao longo da montagem.

### 1.3.3 Algoritmo de Máxima Transferência de Potência

A terceira etapa do projeto será a busca pelo algoritmo ideal de máxima transferência de potência. O objetivo dessa etapa consiste na elaboração de um algoritmo na linguagem de programação C que analise os parâmetros de saída do painel solar e o force a entrar em sua condição de máxima transferência de potência. Testes em laboratório serão fundamentais para a análise da eficiência do algoritmo em um sistema embarcado.

### 1.3.4 Teste de Protótipo Solar via MPPT

A quarta e última etapa de projeto será a realização de testes no protótipo em um sistema real. Após a implementação completa do projeto, será fundamental analisar o protótipo em condições reais de funcionamento para coleta de dados e, de fato, para analisar a viabilidade e a eficiência do sistema solar funcionando via máxima transferência de potência.

## 2 Embasamento Teórico

### 2.1 Paineis Solares

#### 2.1.1 Modelo para Painéis Fotovoltaicos

Ao projetar-se um painel fotovoltaico é fundamental realizar a previsão de parâmetros importantes como irradiação solar, temperatura e carga atendida para o correto dimensionamento de seus componentes. A maioria dos métodos presentes na literatura para calcular-se os parâmetros necessários ao modelo de painéis solares são baseados na medida da curva I-V<sup>1</sup> ou de outros métodos empíricos de realização.

A curva geral I-V de um PV<sup>2</sup> é baseada no modelo exponencial (SERA; TEODOR-RESCU; RODRIGUES, 2007) da seguinte maneira:

$$i = I_{ph} - I_o \left( e^{\frac{v+iR_s}{n_s V_t}} - 1 \right) - \frac{v + iR_s}{R_{sh}}, \quad (2.1)$$

Em que  $V_t$  é a tensão termodinâmica:

$$V_t = \frac{AkT_{stc}}{q} \quad (2.2)$$

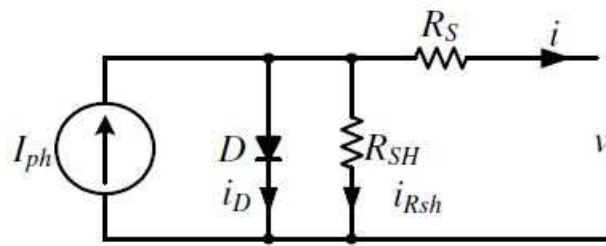


Figura 6 – Sistema equivalente de uma célula fotovoltaica com modelo exponencial

Onde:

- $I_{ph}$  - Corrente gerada via fotopainéis
- $I_o$  - Corrente de saturação
- $R_s$  - Resistência em série do painel

<sup>1</sup> Corrente x Tensão

<sup>2</sup> Photovoltaics -termo recorrente na literatura mundial para sistema fotovoltaico.

- $R_{sh}$  - Resistência em paralelo (Resistor Shunt)
- $A$  - Fator de qualidade do diodo

Faz-se necessário calcular os parâmetros equivalentes ao sistema. Em artigo publicado por Sera, Teodorescu e Rodrigues ([SERA; TEODORESCU; RODRIGUES, 2007](#)) os autores utilizam-se de critérios empíricos e de observação dos dados de fabricantes de materiais para chegar a valores aproximados para os parâmetros. Serão utilizados os valores retirados de seu trabalho para efetuar simulações via software livre de elaboração de circuitos LTSPICE. Para maiores detalhes de como os autores citados chegaram aos referentes valores faz-se importante a leitura na íntegra do artigo.

### 2.1.2 Simulações do Modelo Fotovoltaico

O modelo de painel fotovoltaico foi simulado via software LTSpice acrescido de um modelo simplificado de bateria, no qual a bateria pode ser modelada por uma fonte de tensão e um resistor em série. Adotou-se uma resistência em série muito baixa de forma a simular o comportamento do painel solar com uma carga. Segue o esquemático proposto:

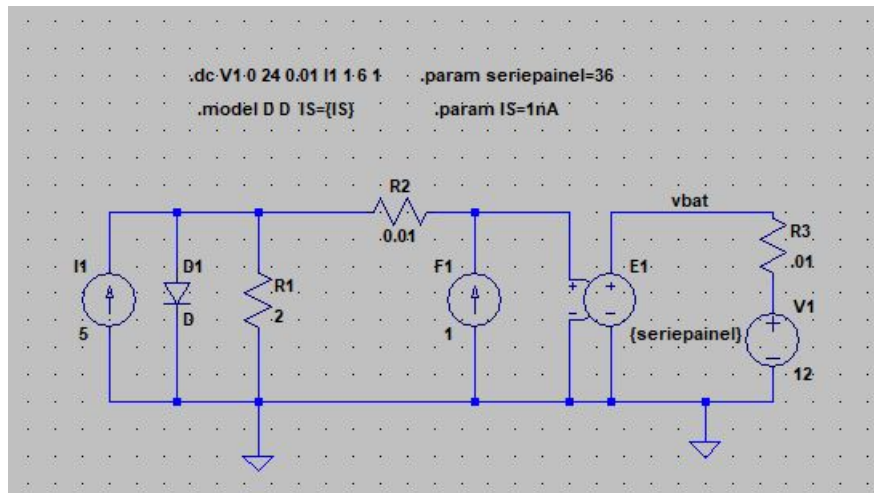


Figura 7 – Modelo de painel solar e bateria simplificados



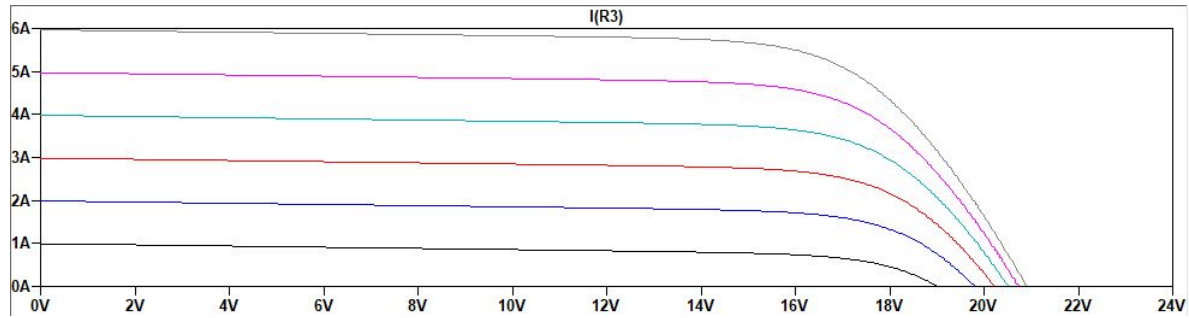


Figura 8 – Simulação (Tensão bateria Vs Corrente de entrada)

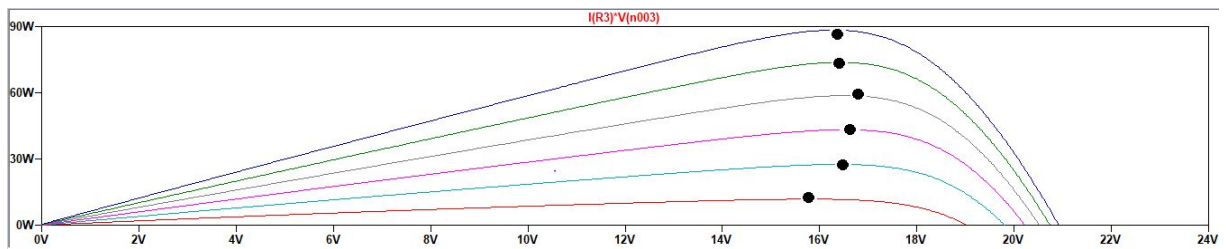


Figura 9 – Simulação (Tensão bateria Vs Potência Entregue a Bateria)

As curvas dispostas em diferentes cores nos gráficos acima simulam irradiações solares distintas. A medida que a irradiação solar aumenta a curva característica relativa ao painel solar tende a descolar-se para cima de forma a aumentar a corrente fluindo sobre seus terminais. Foi acrescentada uma fonte linear de corrente dependente para simular o acréscimo de mais painéis solares ligados em série. Simulando via *DC sweep*, observada-se a relação entre correntes de entrada e tensão na saída do painel (Figura 8). Como a corrente na saída do painel solar depende da irradiação solar, obtemos uma gama de curvas em Figura 8.

Ao analisar-se os resultados da simulação fica evidente que o ponto máximo de potência simbolizado por pontos pretos na (Figura 9) varia de acordo com a tensão na saída do painel solar. A partir dos resultados, constata-se a necessidade de trabalhar-se sempre no ponto de máxima potência.

### 2.1.3 Ponto Máximo de Potência

Observando-se o gráfico da simulação relativo a Figura 9 podemos analisar a potência na saída do painel solar. Se a saída do painel for ligada a uma bateria diretamente sem o auxílio de um controlador de carga, a saída ficará sobre a tensão da bateria. Se a bateria estiver em uma tensão abaixo ou acima do ponto de máxima potência observaremos curvas como as expostas em Figura 10. Evidencia-se a baixa eficiência ao ligar-se diretamente uma célula fotovoltaica a uma bateria, pois a célula dificilmente estará funcionando em seu ponto ótimo nessas circunstâncias.

A Figura 11 simula o ponto de operação do painel solar quando exposto a uma carga resistiva, sendo que o ponto de encontro entre a curva de carga e a curva característica do painel solar delimita o ponto de operação do sistema. É possível observar que a linha de carga pode não encontrar a curva característica do painel solar no ponto de máximo e consequentemente o painel pode não operar no seu limite máximo de potência. A partir do gráfico da Figura 11 fica evidente a relevância do controle sobre a impedância vista pelo painel solar, se a reta de carga não for correspondente com o ponto máximo do painel haverá uma baixa eficiência em relação ao real potencial do sistema.

A área em azul na Figura 10 é descrita como a potência entregue a bateria pelo painel solar. Para atingir-se a potência máxima faz-se necessário que a tensão e a corrente na saída do painel se ajustem para satisfazer a condição de pico de potência. É nesse ponto que o controlador de carga entra em cena, buscando balancear a tensão vista pela célula fotovoltaica e forçando o painel a funcionar em seu ponto de máxima transferência. O controle da tensão pelo painel é feito utilizando conversores DC/DC em conjunto com um processo de controle capaz de ajustar o ponto de operação do painel solar.

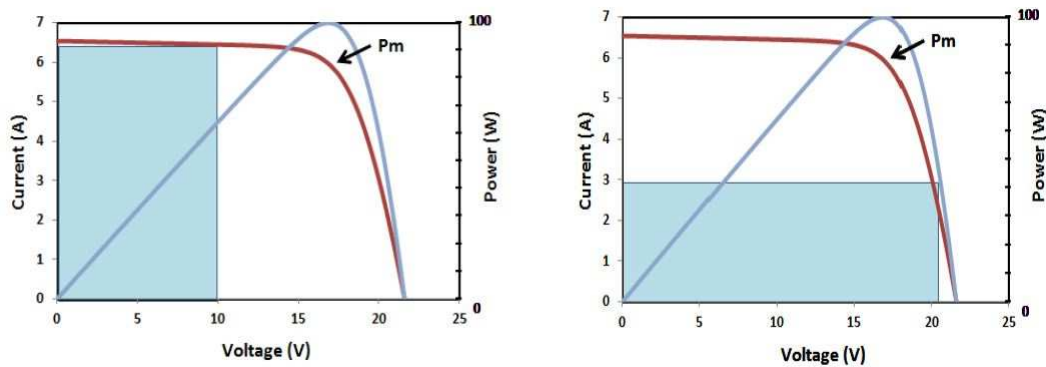


Figura 10 – Tensão bateria Vs Corrente Painel Solar. FONTE: (B.V, 2014)

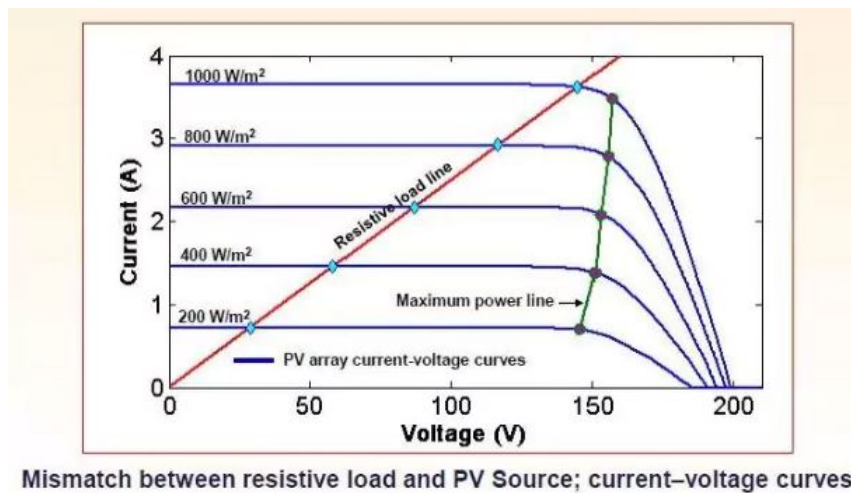


Figura 11 – Tensão Sobre Carga Resistiva Vs Corrente Painel Solar. FONTE: (B.V, 2014)

## 2.2 Conversores DC-DC

O conversor DC/DC é o dispositivo eletrônico responsável pela conversão de tensão contínua para diferentes valores de amplitudes. Os conversores podem ser implementados de diversas formas, sendo o método chaveado um dos mais eficientes se comparado ao método linear, em que resistores simples são usados para controlar a tensão de saída. O método chaveado consiste em transistores controlados via PWM exercendo função de chave que abre e fecha. Se ajustarmos o *duty cycle* das chaves, obtemos uma relação entre a tensão de entrada e a tensão de saída do sistema. Serão analisadas algumas diferentes topologias que podem ser úteis ao longo do projeto.

### 2.2.1 Conversor *Buck*

O conversor *buck* (abaixador) tem como característica a redução da tensão de entrada, ou seja, a tensão de saída é menor que a de entrada. Segue abaixo a topologia do sistema:

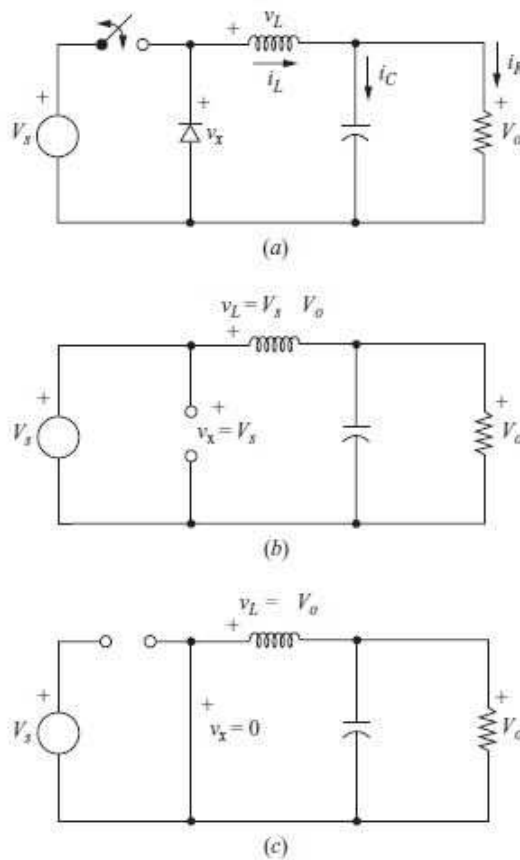


Figura 12 – Conversor *BUCK* (abaixador) FONTE:([HARTS, 2011](#))

O sistema acima possui dois estados básicos, sendo o primeiro com a chave aberta e o segundo com a chave fechada. Vamos analisar os dois estados diante de algumas

suposições:

- O funcionamento do circuito está no estado estável.
- A corrente no indutor está no modo contínuo (sempre positiva).
- O capacitor tem valor bem alto.
- O período de chaveamento é dado como sendo  $T$ ; a chave é aberta por um período  $D.T$  e fechado por um período  $(1-D)T$
- Os componente tem caráter ideal.
- A corrente no indutor é periódica.

$$I_L(t + T) = i_L(t) \quad (2.3)$$

- A corrente média no indutor é zero.

$$V_L = \frac{1}{T} \int_t^{t+T} V_L(\lambda) d\lambda = 0 \quad (2.4)$$

- A corrente média no capacitor é zero.

$$I_c = \frac{1}{T} \int_t^{t+T} i_c(\lambda) d\lambda = 0 \quad (2.5)$$

- A potência fornecida pela fonte é a mesma potência fornecida para a carga.

Abaixo observa-se o comportamento do conversor BUCK, sendo que a Figura 13.a representa a forma de onda da tensão no indutor e a Figura 13.b representa a corrente no indutor e, finalmente, a Figura 13.c representa corrente no capacitor.

- Análise com chave fechada:

$$V_L = V_s - V_o = L \frac{di_L}{dt} \quad (2.6)$$

então,

$$\begin{aligned} \frac{di_L}{dt} &= \frac{V_s - v_o}{L} \\ \frac{di_L}{dt} &= \frac{\Delta i_L}{\Delta t} = \frac{\Delta i_L}{DT} = \frac{V_s - V_o}{L} \end{aligned} \quad (2.7)$$

Chegamos, finalmente, a relação

$$(\Delta i_L)_{fechado} = \left( \frac{V_s - V_o}{L} \right) DT$$

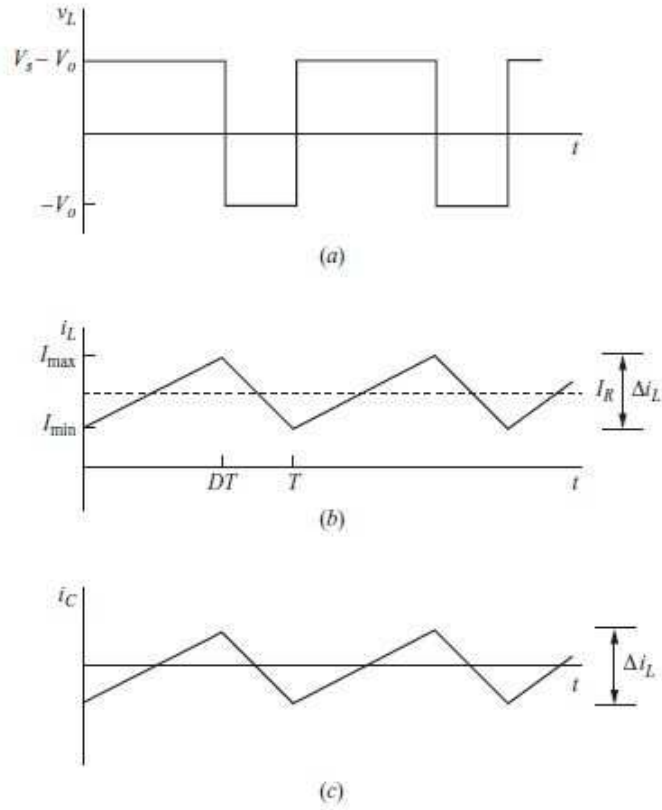


Figura 13 – Forma de onda conversor *BUCK* (abaixador) FONTE:([HARTS, 2011](#))

- Análise para chave aberta:

$$V_L = -V_o = L \frac{di_L}{dt}$$

Ajustando,

$$\frac{di_L}{dt} = \frac{-V_o}{L} \quad (2.8)$$

assim,

$$\begin{aligned} \frac{\Delta i_L}{\Delta t} &= \frac{\Delta i_L}{(1-D)T} = -\frac{V_o}{L} \\ (\Delta i_L)_{aberta} &= -\frac{V_o}{L}(1-D)T \end{aligned} \quad (2.9)$$

No estado estável a troca líquida de corrente no indutor sobre um período é zero:

$$\begin{aligned} \left(\frac{V_s - V_o}{L}\right)DT - \left(\frac{V_o}{L}(1-D)\right)T &= 0 \\ V_o &= V_s D \end{aligned} \quad (2.10)$$

Como o *duty cycle* é um valor de 0 a 1, esse circuito só pode ser usado para abaixar a tensão. Conhecendo os limites inferiores e superiores da corrente no indutor, podemos analisar os parâmetros de indutância com a seguinte abordagem:

$$I_{max} = I_L + \frac{\Delta i_L}{2} = V_o \left( \frac{1}{R} + \frac{1-D}{2Lf} \right) \quad (2.11)$$

Onde,  $L$  é o valor de indutância e  $f$  é a frequência do chaveamento. Para o modo contínuo:

$$I_{min} = 0 = V_o \left( \frac{1}{R} - \frac{1-D}{2Lf} \right) \quad (2.12)$$

$$L \geq \frac{(1-D)R}{2} \quad (2.13)$$

- Tensão de ondulação na saída

Vamos agora dimensionar o capacitor regulador de tensão. Sendo a corrente do capacitor dada por

$$i_c = i_L - i_R$$

e pela definição de capacitância

$$\Delta V_o = \frac{\Delta Q}{C}$$

A variação na carga pode, então, ser expressa

$$\Delta Q = \frac{T \Delta i_L}{8}$$

Usando a Equação 2.9, obtemos

$$\Delta V_o = \frac{TV_o}{8CL}(1-D)T = \frac{V_o(1-D)}{8LCf^2} \quad (2.14)$$

Rearranjando os termos,

$$C = \frac{1-D}{8L \left( \frac{\Delta V_o}{V_o} \right) f^2} \quad (2.15)$$

assim, se faz possível estimar uma capacitância razoável para nossas pretensões.

### 2.2.2 Conversor *Boost*

O conversor boost ou conversor elevador cumpre função oposta ao conversor anterior e eleva a tensão de saída. Usando as mesmas considerações realizadas no conversor BUCK (2.2.1) serão expostas as equações mais importantes sem, todavia, demonstrar cada passo pois a ideia segue a mesma. Sugiro ao leitor sem muita familiaridade com os conceitos de eletrônica de potência o livro do autor Daniel W. Hart principal referência para a base teórica de conversores DC-DC (HARTS, 2011). Segue abaixo o circuito para o conversor *boost*:

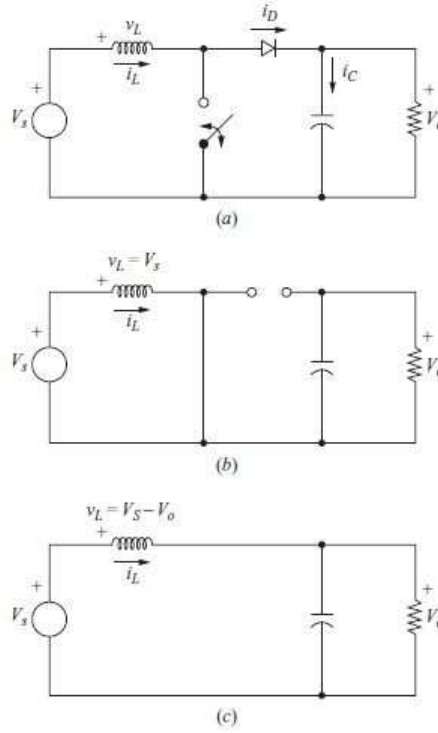


Figura 14 – Forma de onda conversor *boost* (elevador) FONTE:([HARTS, 2011](#))

- Análise para chave fechada:

A partir da mesma análise anterior, chega-se a seguinte equação para variação de corrente no indutor quando a chave está fechada:

$$(\Delta i_L)_{fechada} = \frac{V_s DT}{L} \quad (2.16)$$

- Análise para chave aberta

$$(\Delta i_L)_{aberta} = \frac{(V_s - V_o)(1 - D)T}{L} \quad (2.17)$$

Para funcionamento no estado estável, a variação líquida de corrente no indutor é zero.

$$\frac{V_s DT}{L} + \frac{(V_s - V_o)(1 - D)T}{L} = 0 \quad (2.18)$$

$$V_o = \frac{V_s}{1 - D} \quad (2.19)$$

Como  $D$  varia de 0 a 1 o circuito eleva a tensão de saída. No modo contínuo a indutância é descrita como:

$$L_{min} = \frac{D(1 - D)^2 R}{2f} \quad (2.20)$$

E para tensão de oscilação na saída:

$$C = \frac{D}{R(\frac{\Delta V_o}{V_o})f} \quad (2.21)$$

### 2.2.3 Conversor Buck-Boost

Outro modelo básico de conversor DC/DC é o *buck-boost*, sendo ele a junção dos dois anteriores. Segue abaixo a configuração *buck-boost*:

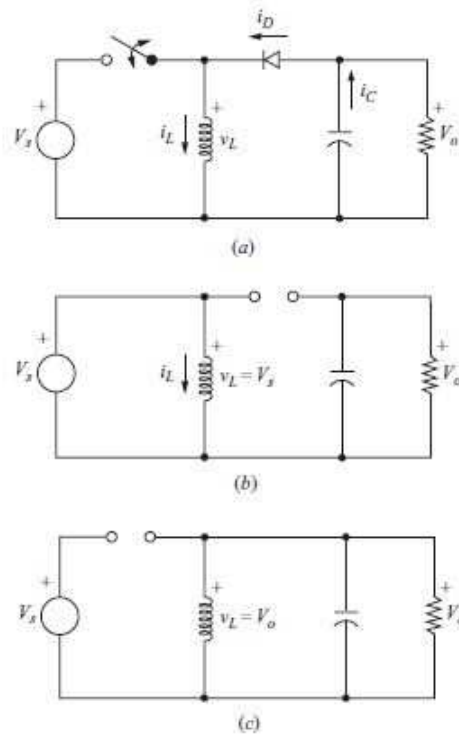


Figura 15 – Forma de onda conversor buck-boost FONTE:([HARTS, 2011](#))

Novamente, partindo do princípio de que as suposições da Seção 2.2.1 são respeitadas será feita a análise para chave aberta e fechada.

- Análise para chave fechada

$$(\Delta i_L)_{fechada} = \frac{V_s DT}{L} \quad (2.22)$$

- Análise para chave aberta

$$(\Delta i_L)_{aberta} = \frac{V_o(1-D)T}{L} \quad (2.23)$$

Como a variação líquida de corrente no indutor é zero sobre um período, então usando as equações 2.22 e 2.23 podemos afirmar que:



$$\frac{V_s DT}{L} + \frac{V_o(1-D)T}{L} = 0 \quad (2.24)$$

$$V_o = -V_s \frac{D}{1-D} \quad (2.25)$$

Consequentemente o *duty cycle* é expressa como sendo,

$$D = \frac{|V_o|}{V_s + |V_o|} \quad (2.26)$$

A indutância mínima será dada analisando, novamente, o fluxo de corrente. Sendo  $L_{min}$  dado por:

$$L_{min} = \frac{(1-D)^2 R}{2f} \quad (2.27)$$

E tensão de ondulação,

$$\frac{\Delta V_o}{V_o} = \frac{D}{RCf}$$

Rearranjando a equação temos:

$$C = \frac{D}{Rf \frac{\Delta V_o}{V_o}} \quad (2.28)$$

Como é fácil observar pela Equação 2.25 a tensão de saída tem polaridade oposta a de entrada. A magnitude da tensão de saída pode, então, ser controlada pelo *duty cycle* e atingir valores maiores ou menores de acordo com o valor da variável D.

#### 2.2.4 Conversor com Indutância Simples no Primário (*SEPIC*)

Através da análise do conversor Buck-Boost observamos que a tensão de saída tem polaridade invertida em relação à de entrada e que para algumas aplicações tal característica pode ser um empecilho e, portanto, indesejada. A partir disso surge a análise do conversor com indutância simples no primário chamado popularmente de *SEPIC*. Tal conversor tem comportamento caracterizado por elevar ou abaixar a tensão de entrada, mantendo a polaridade de entrada. A Figura 16 explicita o circuito do conversor.

Para derivar as relações serão feitas algumas suposições importantes: os valores dos dois indutores são muito altos e suas correntes constantes; os dois capacitores tem valores muito altos e tensões constantes. O circuito está, novamente, funcionando no estado estável e consequentemente as formas de tensão e corrente são constantes. Podemos representar a relação de entrada e saída como sendo,

$$V_o = V_s \left( \frac{D}{1-D} \right) \quad (2.29)$$

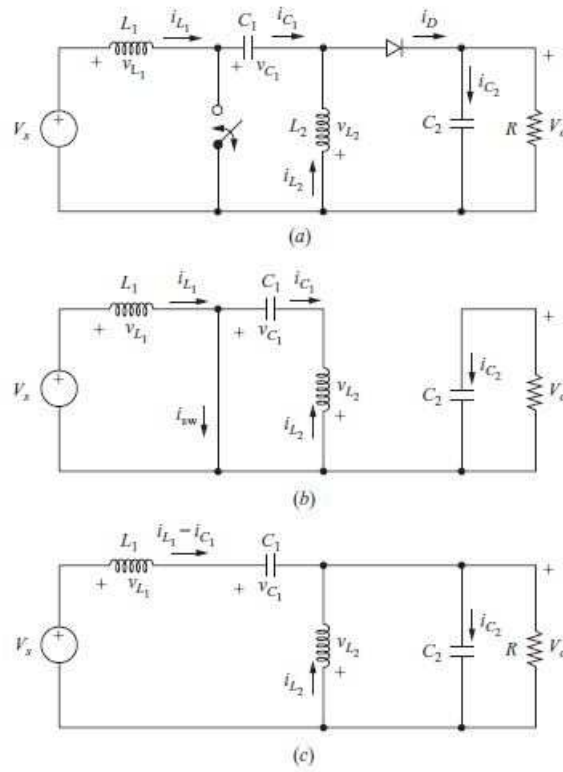


Figura 16 – Forma de onda conversor *SEPIC* FONTE: (HARTS, 2011)

Supondo que não haja perdas no conversor, a potência fornecida pela fonte é a mesma potência absorvida pela carga, ou seja  $P_s = P_o$  que resulta na seguinte relação,

$$P_s I_{L1} = V_o I_o$$

Resolvendo a corrente média no indutor que é também a corrente média na fonte,

$$V_{L1} = V_s = L_1 \left( \frac{\Delta i_{L1}}{\Delta t} \right) \quad (2.30)$$

Resolvendo para  $L_i$ ,

$$\Delta i_{L1} = \frac{V_s D T}{L1} = \frac{V_s D}{L1 f} \quad (2.31)$$

Aplicando a mesma abordagem para  $L_2$  obtemos,

$$\Delta i_{L2} = \frac{V_s D T}{L2} = \frac{V_s D}{L2 f} \quad (2.32)$$

Para os valores de capacitores segue a relação:

$$C_1 = \frac{D}{R(\Delta V_{c1}/V_o)f} \quad (2.33)$$

## 2.3 Microcontroladores

Ao avaliar-se as diretrizes do trabalho, evidencia-se a necessidade de controle dos parâmetros do sistema de captação solar. O controle da corrente e da tensão de saída do painel são de suma importância no algoritmo de máxima transferência de potência, sendo responsável pelo valor ideal do *duty cycle* empregada no controle dos conversores de corrente contínua. A avaliação de corrente e da tensão de entrada será feita por meio de microcontrolador que é o responsável por verificar os parâmetros de entrada e decidir pela melhor saída possível em termos de *duty cycle*. O microcontrolador tem por objetivo, portanto, executar o algoritmo de máxima transferência de potência e garantir a máxima eficiência do sistema. O microcontrolador escolhido para a implementação do protótipo foi o MSP430 fabricado pela Texas Instrumentos. No controle de corrente e tensão do sistema de saída faz-se necessário o uso de componentes básicos presentes no microcontrolador como temporizadores e conversores analógicos/digitais. Introduziremos uma análise teórica sobre as principais ferramentas utilizadas no processo.

### 2.3.1 MSP430

O MSP430 é um microcontrolador de 16-bit de baixo custo construído para aplicações em sistemas embarcados destacando-se pela característica de baixo consumo de energia. O modelo MSP430F5529 será usado para a implementação de projeto. Seguem as especificações do microcontrolador na (Tabela 3).



Figura 17 – Modelo MSP430 Texas Instrumentos. FONTE:([TI, 2008](#))

Tabela 1 – Parâmetros MSP430F5529

CPU	MSP430
Máxima Frequência	25MHz
Memória não Volátil	128 KB
GPIO Pins	63
ADC	ADC12 -14ch
Timers	4
Vcc	[1,8 V até 3,6 V]

### 2.3.2 Modulação por Largura de Pulso

Modulação por largura de pulso mais comumente conhecido por sua sigla em inglês PWM (Pulse Width Modulation) é uma técnica de modulação em que uma geração de pulsos digitais periódicos com largura controlável pode ser usada para diversos fins. Uma aplicação muito utilizada é no controle de tensão e potência entregue à carga. No caso em questão, a modulação por largura de pulso será a responsável pelo controle da tensão do conversor de tensão. Será ilustrado esse princípio através de um conversor chaveado básico mostrado na Figura 18.

A componente média de tensão pode ser calculada através da seguinte relação:

$$V_o = \frac{1}{T} \int_0^T V_o(t) dt = \frac{1}{T} \int_0^{DT} V_s(t) dt = V_s D \quad (2.34)$$

A componente DC da tensão sobre a carga fica então uma função de D que pode ser representado como sendo,

$$D = \frac{t_{ligado}}{t_{ligado} + t_{desligado}} = \frac{t_{ligado}}{T} = T_{ligado} f \quad (2.35)$$

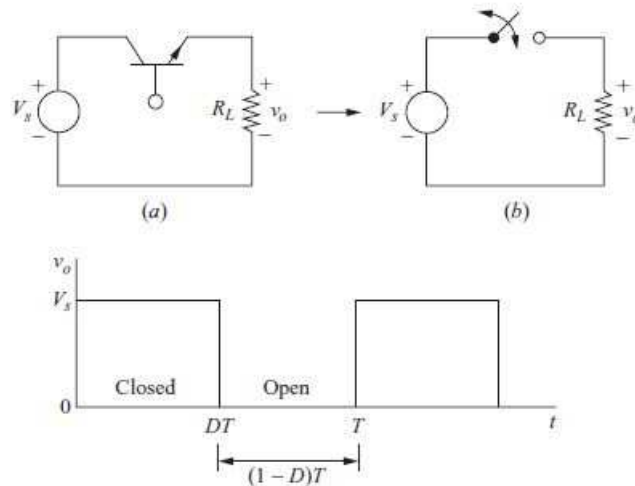


Figura 18 – PWM em controle de tensão. FONTE:([HARTS, 2011](#))

Destaca-se que o responsável por gerar o pulso por largura de banda será o microcontrolador MSP430.

### 2.3.3 Temporizadores

Temporizadores ou timers são peças importantes em sistemas embarcados. Eles são os responsáveis por gerar um referencial de tempo em aplicações que exigem sincronismo temporal. A forma com que o MSP430 obtém a noção de tempo é por meio da velocidade de clock. O dispositivo que controla a distribuição dos clocks dentro do MSP40 é conhecido como UCS <sup>3</sup> e possui cinco tipos de fonte para sua configuração:

- Cristal 1 – 32768Hz
- Cristal 2 – 4MHz
- VLO – 10KHz
- REFO – 32768kHz
- DCO – Gerado a partir dos cristais

No caso específico do modelo MSP430F5529 existem disponíveis quatro tipos de temporizadores de 16 bits. Para a aplicação específica em projeto, será utilizado o timer A do dispositivo. Seguem abaixo algumas características importantes do dispositivo e seu esquemático (Figura 19):

- 16-bit timer/contador assíncrono com quatro modos de operação.
- Fonte de clock configurável/selecionável.
- Mais de sete modos de configuração para captura e comparação de registradores.
- Saídas configuráveis com PWM
- Vetores de interrupção para análise simples de todas as interrupções relativas aos timers.

---

<sup>3</sup> Sigla para o termo em inglês (*Unified Clock System*)

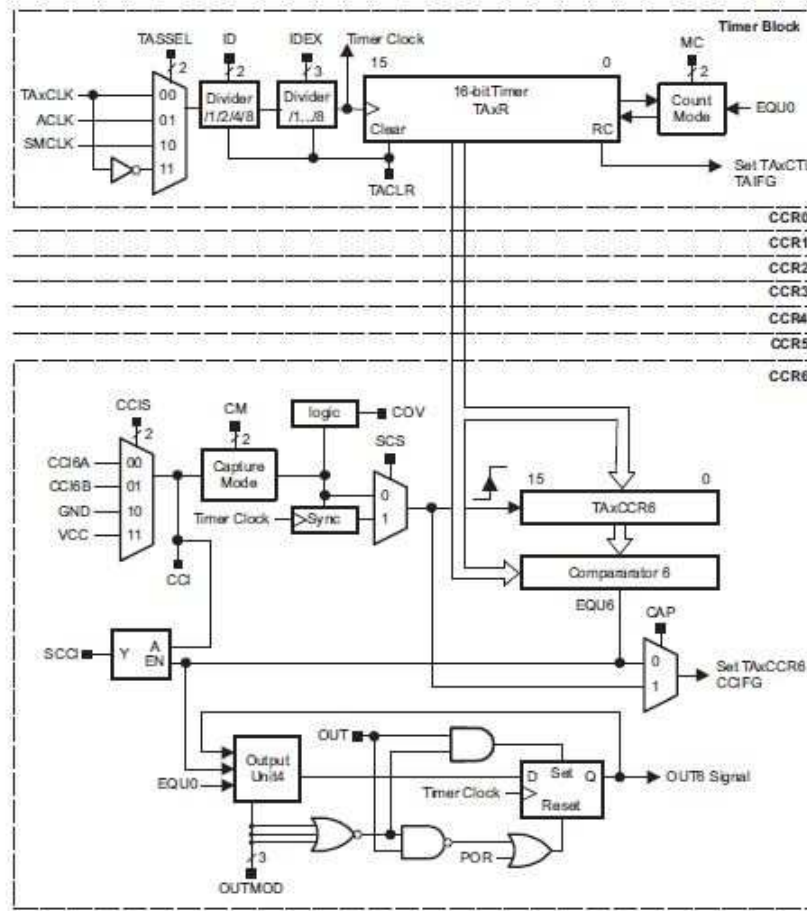


Figura 19 – Esquemático do timer A MSP430F5529.FONTE:([TI, 2008](#))

O sinal PWM será gerado ajustando o registrador CCR0 como teto de contagem e assim definindo a frequência de operação. O *duty cycle* será definido pelo registrador CCR1 que é um limiar intermediário de contagem.

## 2.4 Algoritmo de Máxima Transferência de Potência

### 2.4.1 Perturba e Observa

Como já foi salientado, uma pequena perturbação no sistema pode causar mudança na curva de potência do painel solar como se faz notar na Figura 9. O algoritmo conhecido como P e O (Figura 20) trabalha através da análise de  $V(k)$  e  $I(k)$  sendo esses a tensão e a corrente na iteração atual e  $V(k-1)$  e  $I(k-1)$ , os mesmos valores medidos na iteração anterior. O algoritmo funciona, basicamente, analisando tensão e corrente em instantes discretos de tempo e averiguando a potência do sistema. Se o sistema mostra um aumento de potência durante a perturbação, o algoritmo analisa a tensão atual e a anterior escolhendo então se deve ajustar o duty cycle para aumento ou redução da tensão vista pelo painel. Se o sistema observa uma redução da potência na saída do sistema,

ele analisa novamente a tensão atual e anterior para a escolha sobre o correto valor de duty cycle a ser empregado. Esse algoritmo é bastante utilizado por sua simplicidade de implementação e relativa eficiência.

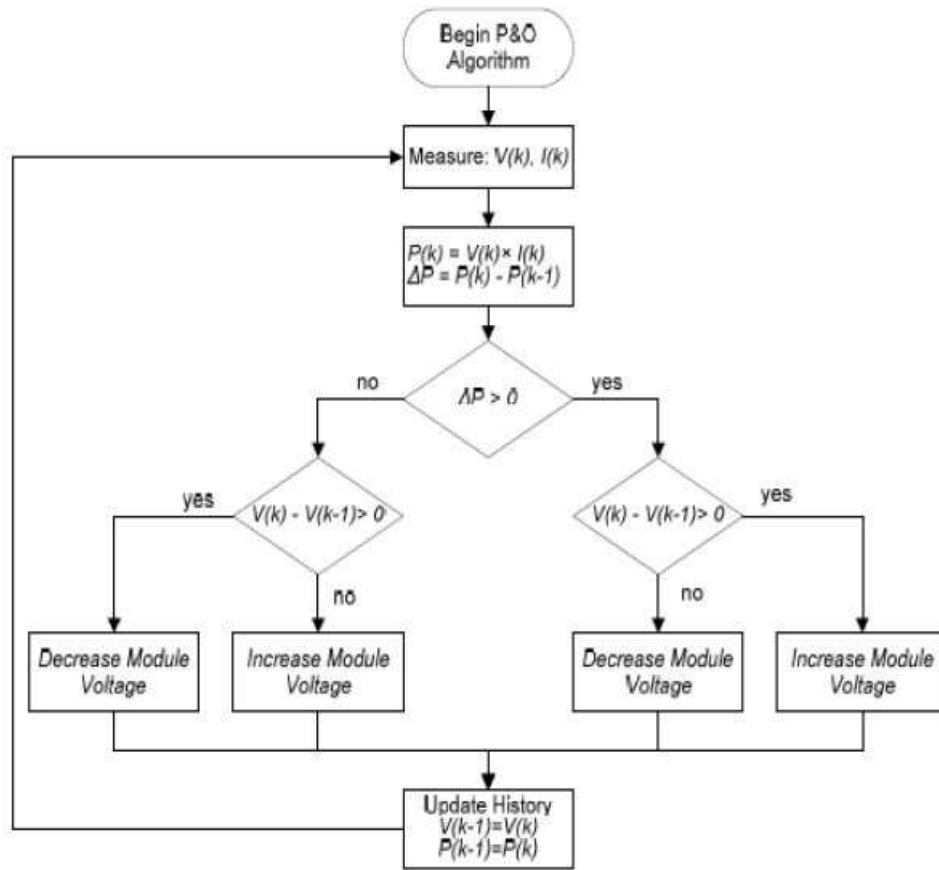


Figura 20 – Algoritmo Perturba e Observa. FONTE:(YADAV1; S.THIRUMALIAH2; G.HARITHA3, 2012)

### 2.4.2 Condutância Incremental

O método CI é também bastante utilizado por sua relativa simplicidade. Se comparado ao método Perturba e Observa ele possui certas vantagens em relação à estabilidade quando valores próximos ao ponto máximo de potência são atingidos. Como pode-se notar na Figura 21 o método CI pode determinar quando o MPPT é atingido e, diferentemente do método anterior, pode parar de perturbar o sistema. O algoritmo calcula a admitância do sistema e sua derivada para depois somar os dois termos. Se a soma for igual a zero não é necessário perturbar o sistema alterando o *duty cycle*. Se a soma for negativa, aumenta-se o *duty cycle* e se a soma for positiva diminuimos a taxa até atingir o MPPT.

## Algorithm [7]

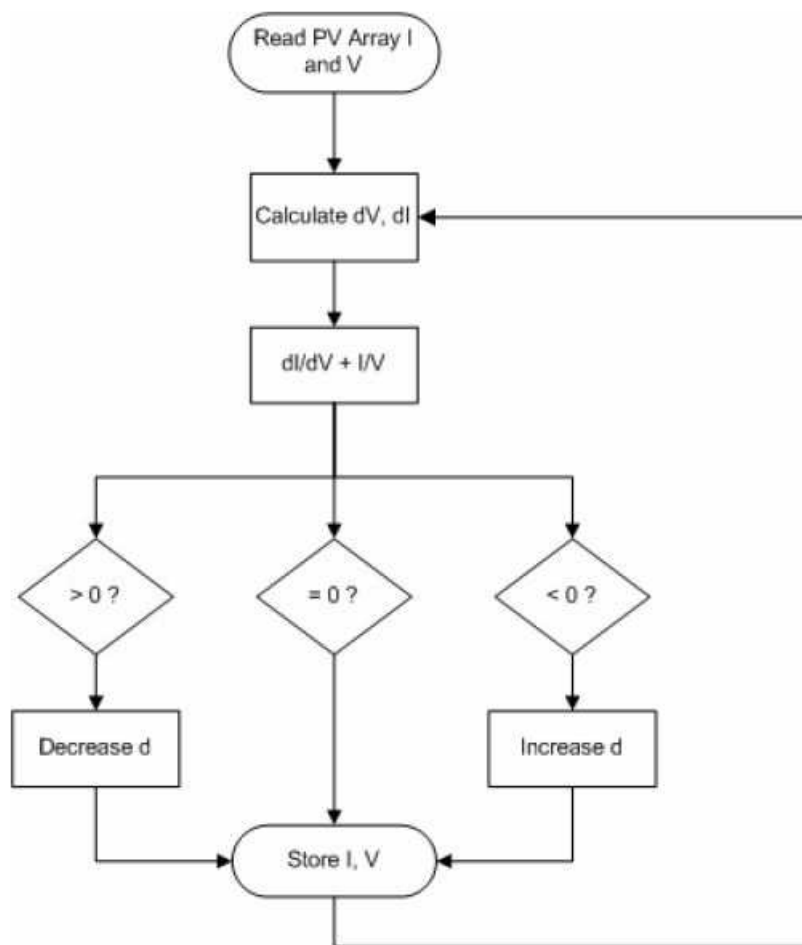


Figura 21 – Algoritmo Condutância Incremental.FONTE:([YADAV1](#); [S.THIRUMALIAH2](#); [G.HARITHA3](#), 2012)



## 3 Metodologia

A ordem metodológica proposta no processo de pesquisa baseou-se primeiramente na busca por referências teóricas e na realização de simulações, sendo essa primeira parte puramente explicativa. Após realizadas as diversas simulações necessárias ao projeto, buscou-se colocar em prática todos os circuitos propostos. A pesquisa tomou, então, caráter empírico exploratório e os resultados qualitativos relativos ao sistema foram amplamente explorados e serão aqui discutidos.

### 3.1 Metodologia de Medidas

#### 3.1.1 Caracterização de Painel Solar

Foram adquiridos para o projeto dois tipos de painéis solares. Um painel de baixa potência para primeiros testes e a verificação do princípio de funcionamento e um segundo painel com maior potência para a efetivação do circuito. Segue abaixo as características dos mesmos:



Figura 22 – Pannel solar 12V 22W

Tabela 2 – Parâmetros Painel Solar 12V 22W

Peso do Módulo	2,26 kg
Vidro	Vidro Temperado 3,2 mm, Silício Policristalino
Máxima Potência (Pm)	22 Watts
Tensão de Máxima Potência (Vm)	17.8 Volts
Corrente de Máxima Potência (Im)	1.24 Amps
Tensão de Circuito Aberto (Voc)	21.7 Volts
Corrente de Curto-Circuito (Isc)	1.35 Amps
Tensão Máxima do Sistema	50 Volts
Dimensões do painel	(360 x 480 x 25) mm



Figura 23 – Painel solar 5V 1W

Tabela 3 – Parâmetros Mini Painel Solar 5V 1W

Tensão Nominal	5V
Corrente Nominal	160 mA
Potência Nominal	1 W
Tamanho	150mm x 69mm
Tensão Máxima do Sistema	50 Volts
Dimensões do painel	(360 x 480 x 25) mm

A caracterização do painel solar [12V 22W] foi feita empiricamente em laboratório. Aplicamos uma luz ambiente ao dispositivo e conseqüentemente quase nenhuma potência foi gerada. Foi imposta uma tensão nos seus terminais com um fonte de tensão DC e anotamos os dados de corrente fluindo sobre o painel solar versus tensão imposta pela fonte. Os resultados dessas medidas serão analisados no Capítulo 4.

### 3.1.2 Conversor/Painel Solar

Primeiramente o circuito *SEPIC* foi simulado via *software* livre LTSPICE 3.2. O sistema simulado usou uma carga resistiva de 50 ohms simulando uma carga e uma tensão fixa de 5V.

Após a simulação (disposta em Seção 4.2.1) o processo foi implementado empiricamente em laboratório considerando um conjunto de pilhas recarregáveis e o mini painel solar. Segue disposto na Tabela 4 os componentes usados para a montagem do circuito SEPIC.

Tabela 4 – Relação de Componentes Utilizados

Componentes	Quantidade	Especificação
Capacitores Eletrolíticos	2	22uF
Capacitores de cerâmica	2	5.6pF
Indutores Radiais	2	680uH
Diodos	1	6 Amps 1000Volts
Transistor CMOS	1	BUZ11 30A 50V rds=0.04ohms
Pilhas Recarregáveis	2	1,2 Volts/2300mAh
Mini Painel solar	1	5 Volts/150mA

Em laboratório o circuito foi testado em conjunto com o painel solar e pilhas recarregáveis com tensão de aproximadamente 2,4V (duas pilhas em série de 1.2V 2300mAh) para que houvesse uma verificação mais apurada de sua capacidade de alterar, de fato, o ponto de operação do painel solar.

Alteramos o *duty cycle* do sistema para então colher dados, dispostos na Seção 4.2.2, referentes à corrente de saída do painel solar e à tensão entre os terminais do sistema solar. A corrente fluindo sobre os terminais do painel solar foi medida através da alocação de um multímetro em série com o painel. Um multímetro em paralelo ao painel foi alocada para retirada de dados relativos a variação da tensão. A luz solar foi simulada com o uso de uma lâmpada incandescente.

### 3.1.3 Sensores

Primeiramente, o circuito a ser implementado, disposto em seção 4.3 na Figura 38, foi devidamente simulado para posteriormente ser montado em laboratório. A lista com os componentes usados para a montagem do sensor de corrente pode ser vista em Tabela 5:

Tabela 5 – Relação de Componentes Utilizados

Componentes	Quantidade	Especificação
Resistores de filme de Carbono	10	Diversos Valores
Resistor de fio	1	0.1 Ohms
Amplificadores Operacionais	3	LM358

Para verificar a eficiência do sistema leitor de corrente foi imposta uma quantidade de luz artificial provinda de uma lâmpada incandescente ao painel solar, em conformidade com Figura 30 disposta em seção 4.2. Dada uma intensidade luminosa aproximadamente constante variamos o *duty cycle* da onda PWM gerada via microcontrolador e observamos a tensão na saída do sensor de corrente com o uso de um osciloscópio. Um multímetro foi colocado em série com o painel solar para análise dos dados de corrente média circulando nos terminais do painel solar e outro multímetro foi disposto em paralelo com o painel solar para retirada de dados referentes a variação da tensão sobre o painel solar.

### 3.1.4 Conversão Analógica Digital e Algoritmo P&O

Os dados referentes à conversão dos sinais analógicos para digitais, foram recolhidos a partir do software Code Composer Studio. Foram aplicados os sinais de corrente e tensão em duas portas do microcontrolador e analisados os dados obtidos. Com os dados de corrente e tensão já digitalizados, o algoritmo Perturba e Observa descrito em Seção 2.4 foi aplicado para obter a relação de potência máxima solicitada.

## 3.2 Ferramentas Utilizadas

Segue abaixo uma descrição das principais ferramentas utilizadas ao longo do projeto:

- LTspice XVII(x64) – Software livre de simulação de circuitos eletrônicos. Autor: Mike Engelhardt
- Scilab 6.0.2 (x64) – Software livre usado na obtenção de gráficos e análise de dados colhidos.
- MSP430F5529LP – Microcontrolador utilizado no processo. Na Seção [2.3](#) pode-se observar uma descrição mais detalhada do controlador usado.
- Code Composer Studio (Versão 7) – Software base para programação de microcontroladores desenvolvido pela Texas Instrumentos.



## 4 Resultados e Discussão

### 4.1 Painel Solar

Segue abaixo os dados e a curva obtida do painel com baixíssima luz ambiente de acordo com o processo metodológico descrito em Seção 3.1.1:

Tabela 6 – Relação entre a tensão aplicada e corrente no painel solar

Tensão [V]	Corrente[mA]
10	1.4
11	0.6
11.5	0
12	-0.7
12.5	-1.4
13	-2.3
13.5	-3.4
14	-4.7
14.5	-6.
15	-8.4
15.5	-11
16	-14.
16.5	-18.3
17	-23.7
17.5	-30.9
18	-40.6
18.5	-53.8
19	-71.8
19.5	-96.3
20	-129.4
20.5	-173
21	-229
21.5	-300
21.8	-348
22	-384

O sinal negativo da corrente (Tabela 6) deve-se ao fato de que foi adotado o sentido positivo como sendo a corrente saindo do painel solar. Com o objetivo de se ajustar o eixo das ordenadas para o sentido positivo, o gráfico foi deslocado um ampére para cima e assim simulamos o funcionamento do painel como gerador e não mais como carga. O gráfico obtido, disposto em Figura 24, é coerente com o modelo apresentado e simulado na Seção 2.1.1.

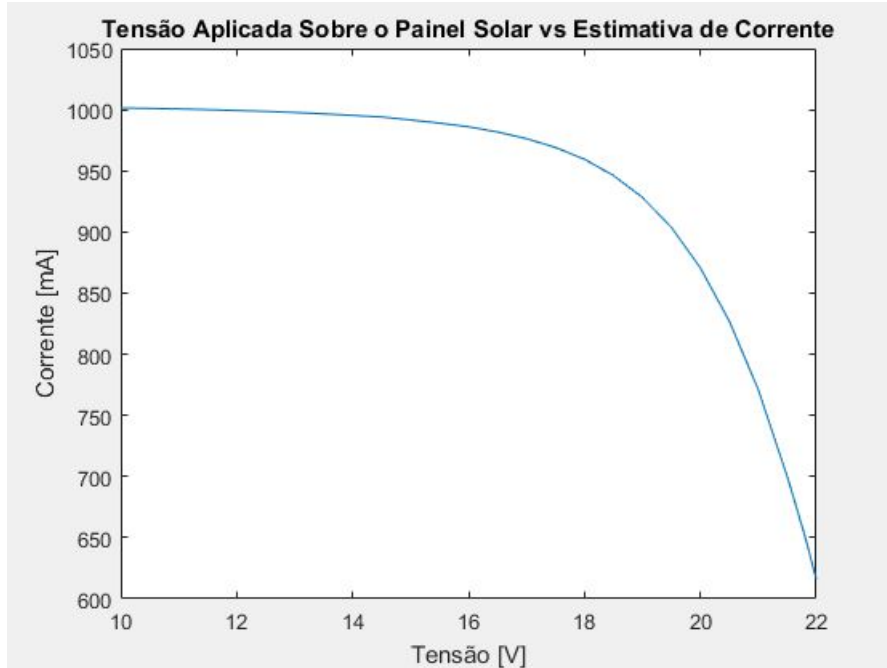


Figura 24 – Gráfico V x I obtido em laboratório

## 4.2 Sistema Conversor/Painel Solar

Ao analisar o tipo de circuito necessário para a conversão DC/DC do controlador foi preciso uma análise qualitativa do problema. A bateria tende a controlar a tensão de saída do circuito conversor (Figura 5), desse modo faz-se necessário um controlador de carga apto a aumentar ou diminuir a tensão vista pelo painel solar de acordo com a proporção necessária. Se a bateria possui tensão em operação menor do que a tensão no ponto máximo do painel solar, um circuito do tipo *buck* é requerido, já que para manter a proporção  $\frac{V_{out}}{V_{in}} < 1$  do sistema faz-se necessário aumentar a tensão observada pelo painel na entrada do circuito. Por outro lado, se a bateria funciona em tensão superior ao ponto máximo de operação do painel solar é necessário um conversor do tipo *boost* para que se mantenha a proporção do sistema como  $\frac{V_{out}}{V_{in}} > 1$ .

Na literatura existente a respeito de controladores é fácil verificar diversos artigos e implementações de circuitos que utilizam as topologias *buck* e *boost*. Dado as diversas referências a esses circuitos conversores decidiu-se analisar o comportamento de um sistema mais genérico do tipo *SEPIC*. Esse circuito é capaz de aumentar ou diminuir a tensão de saída variando a forma de onda via PWM.



### 4.2.1 Simulações do Circuito DC/DC

Abaixo observamos o circuito SEPIC simulado para distintos valores de *duty cycle*. A simulação está em conformidade com o esperado dado que para um  $D < 0,5$  o circuito funciona de fato como um abaixador de tensão e para valores de  $D > 0,5$  o circuito passa a comportar-se com um elevador de tensão com relação saída/entrada.

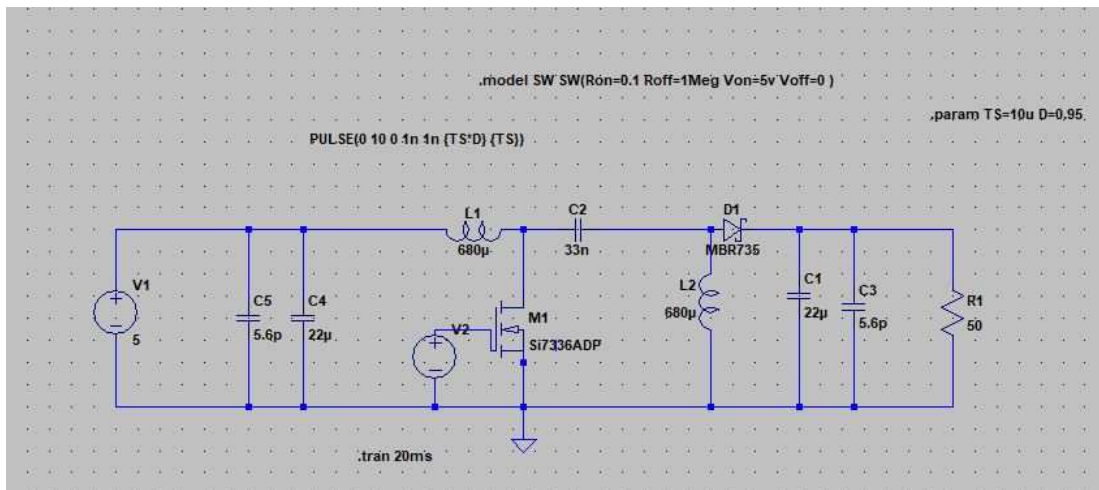


Figura 25 – Topologia SEPIC

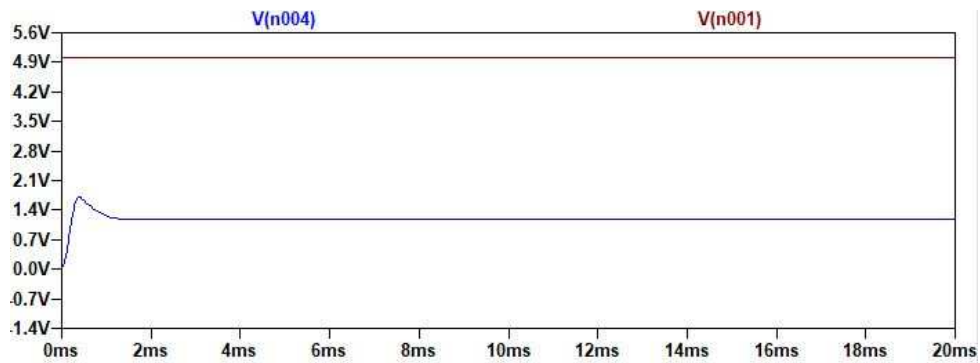


Figura 26 – Simulação  $f=100\text{kHz}$   $D=0.2$

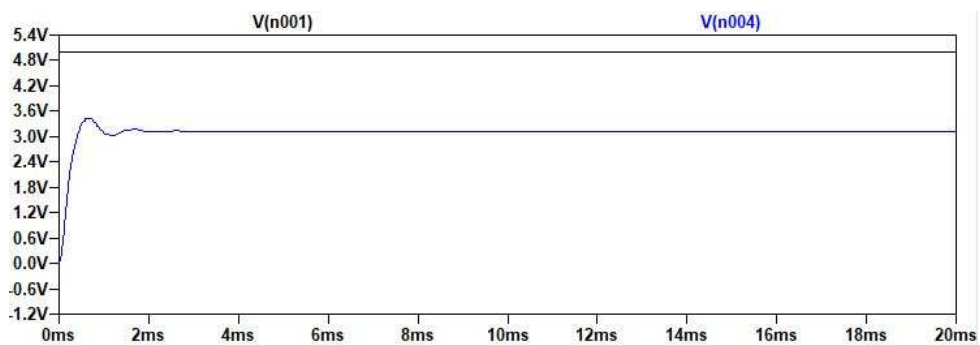


Figura 27 – Simulação Abaixador de Tensão  $f=100\text{kHz}$   $D=0.4$  abaixador

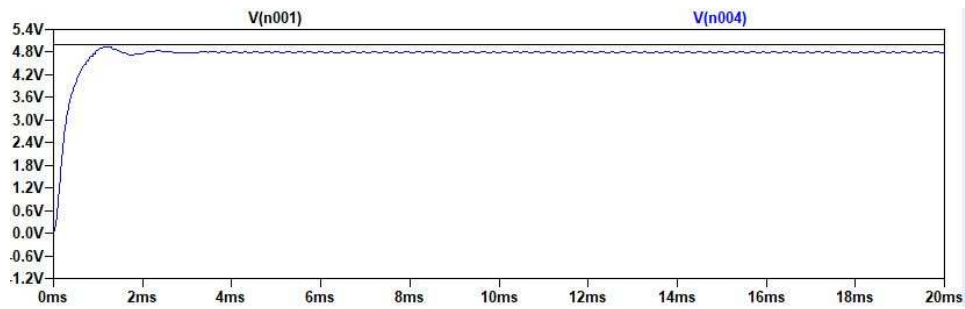


Figura 28 – Simulação Abaixador de Tensão  $f=100\text{kHz}$   $D=0.5$

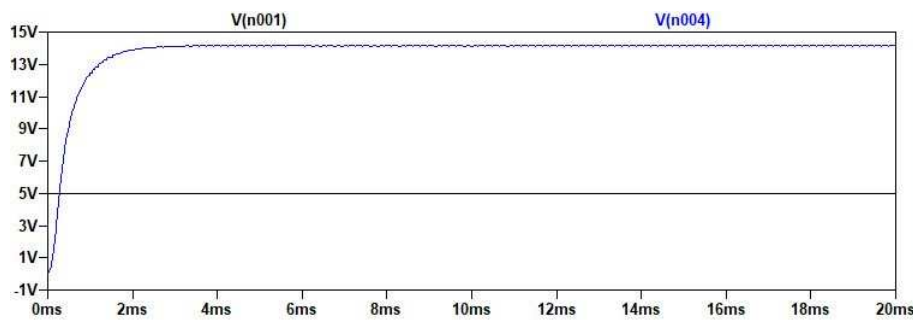


Figura 29 – Simulação Elevador de Tensão  $f=100\text{kHz}$   $D=0.8$

### 4.2.2 Montagem do Sistema DC/DC - Painel - Baterias

Primeiramente o esquemático apresentado na Figura 5 foi montado. O microcontrolador foi o responsável por alterar a onda PWM (100KHz) de acordo com a manipulação dos botões do mesmo, ou seja, apertando o botão direito do microcontrolador aumentamos o *duty cycle* e apertando o botão esquerdo diminui-se o *duty cycle*. O código referente ao microcontrolador encontra-se em Apêndice A.1. O processo metodológico de montagem foi devidamente descrito em Seção 3.1.2. Seguem abaixo imagens dos testes realizados em laboratório:

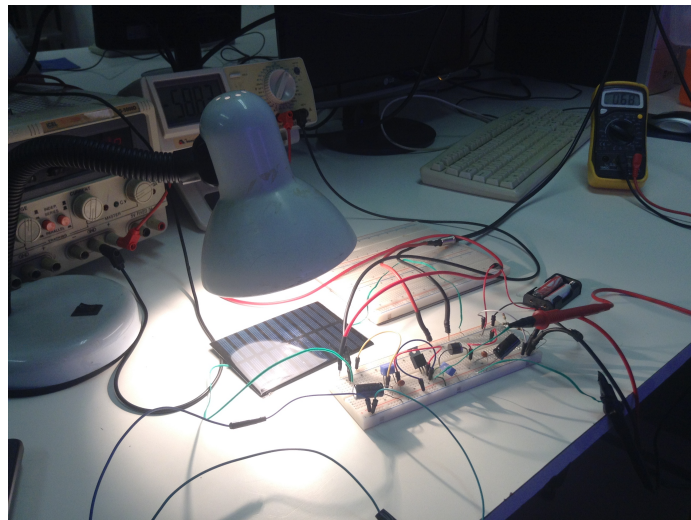


Figura 30 – Montagem Realizada em Laboratório Lab 1 - SG11



Figura 31 – Relação Tensão na Saída do painel vs onda PWM para D=38,2%

Após a coleta de dados do sistema, evidencia-se a relação de correntes e tensões na saída do painel solar e suas variações de acordo o *duty cycle* do circuito DC/DC SEPIC.

Tabela 7 – Relação PWM Vs Potência Extraída do Painel Solar

<i>Duty Cycle</i> (%)	Corrente Média (milliampere)	Tensão Média (Volts)	Potência (milliwatts)
2,6	1,32	5,61	7,4052
7,9	14,13	5,54	78,2802
10,5	25,43	5,20	132,2360
13,2	36,00	4,96	178,5600
15,8	46,30	4,80	222,2400
18,4	55,01	4,50	247,5450
20,4	58,4	4,03	236,8028
27,6	61,65	3,02	195,2430
38,2	62,44	2,24	139,8656
68,4	64,21	1,13	72,5573
96,1	65,35	0,6	39,2100

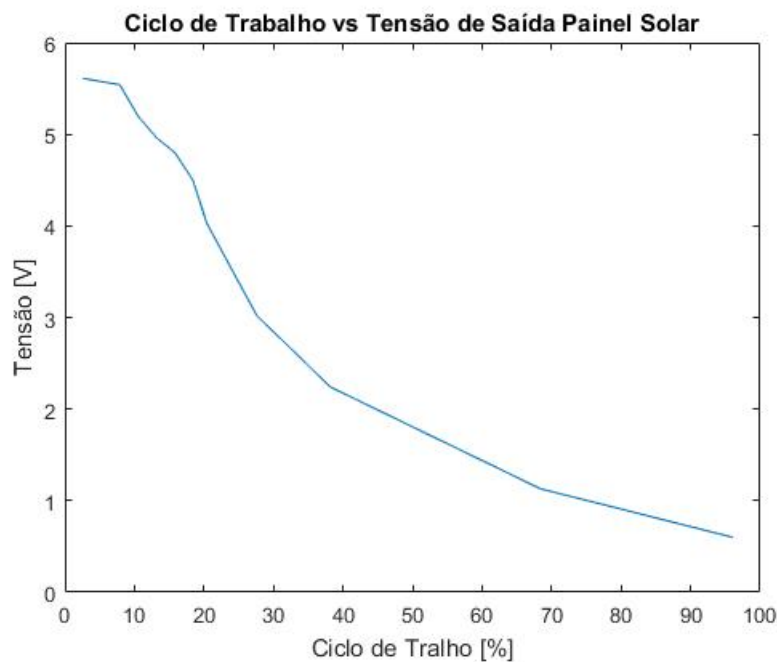


Figura 32 – *Duty Cycle* Vs Tensão de saída no painel

Fica clara a relação de máxima potência extraída do painel solar para uma dada condição de luz ambiente. Para a condição de luz artificial alocada no experimento, a máxima potência gira em torno de 249 mW para um  $D=19,6\%$ ,  $I=58\text{mA}$ ,  $V=4,40\text{V}$  (Figura 35). As pilhas recarregáveis impõem uma tensão de aproximadamente 2,4V na saída do conversor DC/DC, ou seja, uma tensão abaixo do ponto de máxima potência do painel. O conversor SEPIC altera a relação de impedâncias vista pelo painel e, conseqüentemente, a relação de tensão entrada/saída. Para um *duty cycle* menor que 50% deveríamos obter,  $0 < \frac{V_{out}}{V_{in}} \leq 1$  (Equação 2.29). Considerando que  $V_{out}$  é constante e igual a tensão da

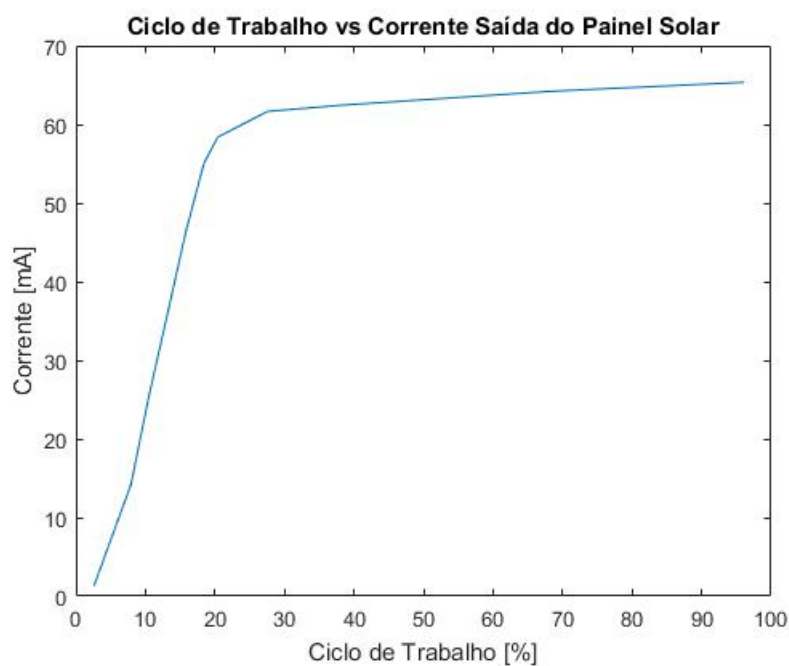


Figura 33 – Gráfico *Duty Cycle* Vs Corrente na saída do painel

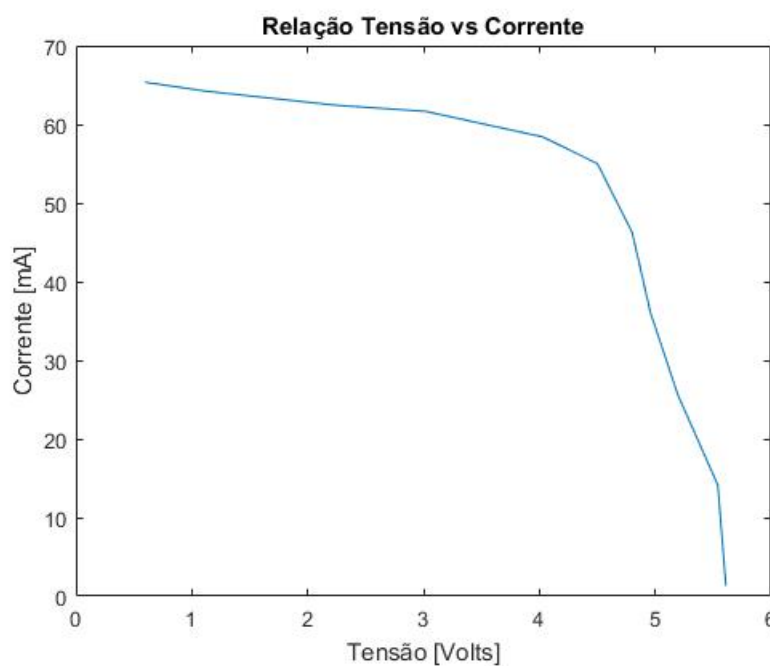


Figura 34 – Tensão Vs Corrente na saída do painel

pilha, a tensão  $V_{in}$  tende a diminuir a medida que o *duty cycle* aumenta para manter a proporção.

Entretanto, foi constatado nos testes realizados em laboratório que a relação do tipo abaixador de tensão só é observada para um *duty cycle* entre 0% até aproximadamente 27%. Tal discrepância pode estar relacionada com a baixa impedância da pilha. Para um

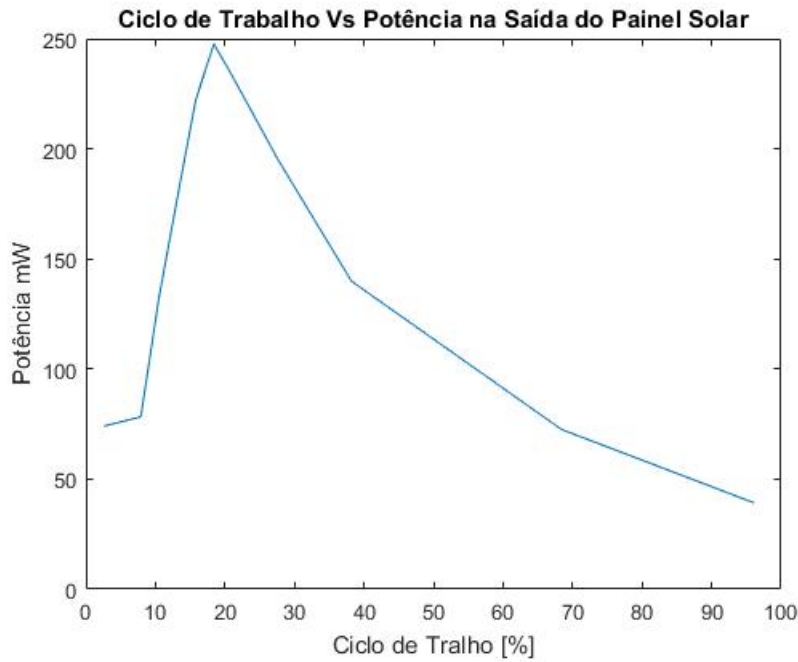


Figura 35 – Gráfico *Duty Cycle* Vs Potência na saída do painel solar

*duty cycle* maior que aproximadamente 30% o circuito passa a ter características de um conversor do *Boost* com a seguinte relação  $\frac{V_{out}}{V_{in}} \geq 1$ .

É importante salientar que a medida que o *duty cycle* aumenta, observa-se nas imagens obtidas via osciloscópio, que a onda PWM afeta a tensão de saída do painel solar fazendo com que a tensão flutue em torno de um valor médio. Esse valor médio foi usado para se traçar as curvas que relacionam *duty cycle* com tensões e correntes.

## 4.3 Sensores

### 4.3.1 Simulações Sensor de Corrente

Para que o algoritmo de máxima transferência de potência seja realizável, é necessário um bom sensor de corrente. A princípio foi pensada a utilização de um sensor de corrente via efeito hall, porém como as correntes do protótipo montado são da ordem de *miliampere*, um sensor do tipo invasivo apresentou-se mais plausível. O sensor montado no projeto foi baseado no sensor *uCurrent™ GOLD*<sup>1</sup> (Figura 37).

Algumas adaptações foram feitas com o objetivo de simplificar o circuito *uCurrent*. É fácil observar a retirada de um amplificador operacional (Figura 38) que funcionava como filtro, assim como a mudança no ganho de tensão em relação ao circuito base. Notamos que o circuito sensor de corrente (Figura 38) possui duas saídas *v\_current* e

<sup>1</sup> ([UCURRENT...](#), )



Figura 36 – Sensor de Corrente

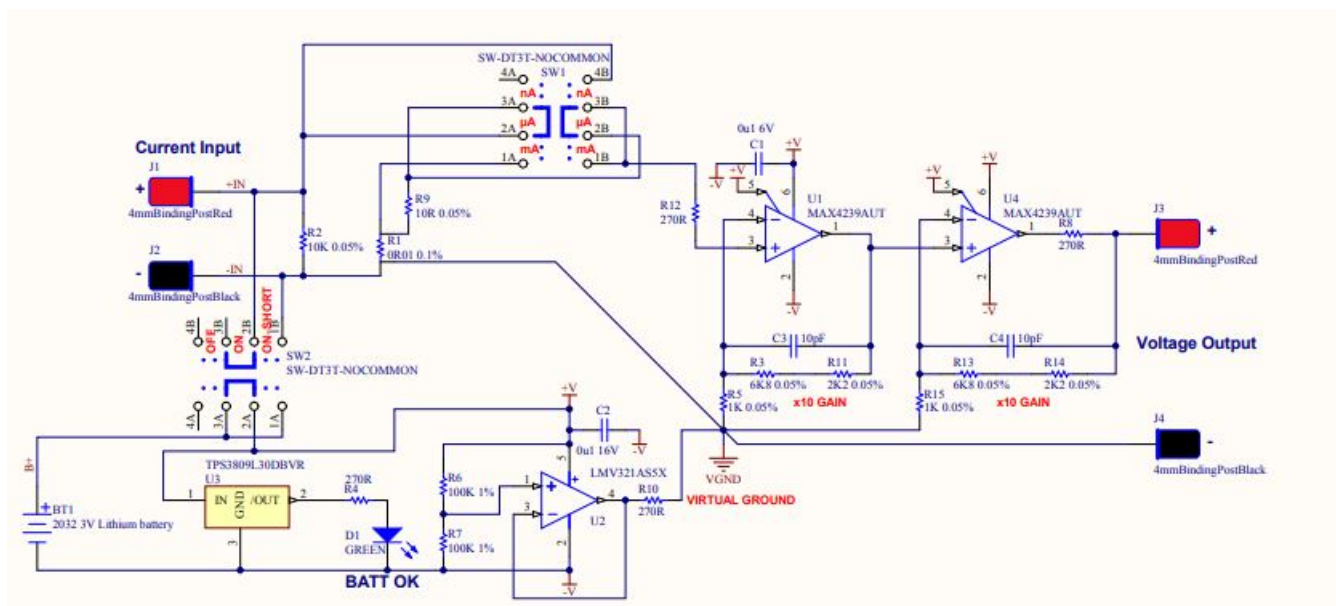


Figura 37 – Circuito de Sensor de Corrente uGold

$v_{ref\_bat}$  e a diferença de tensão entre ambas é proporcional a corrente que passa pelo resistor  $R_{sense}$ .



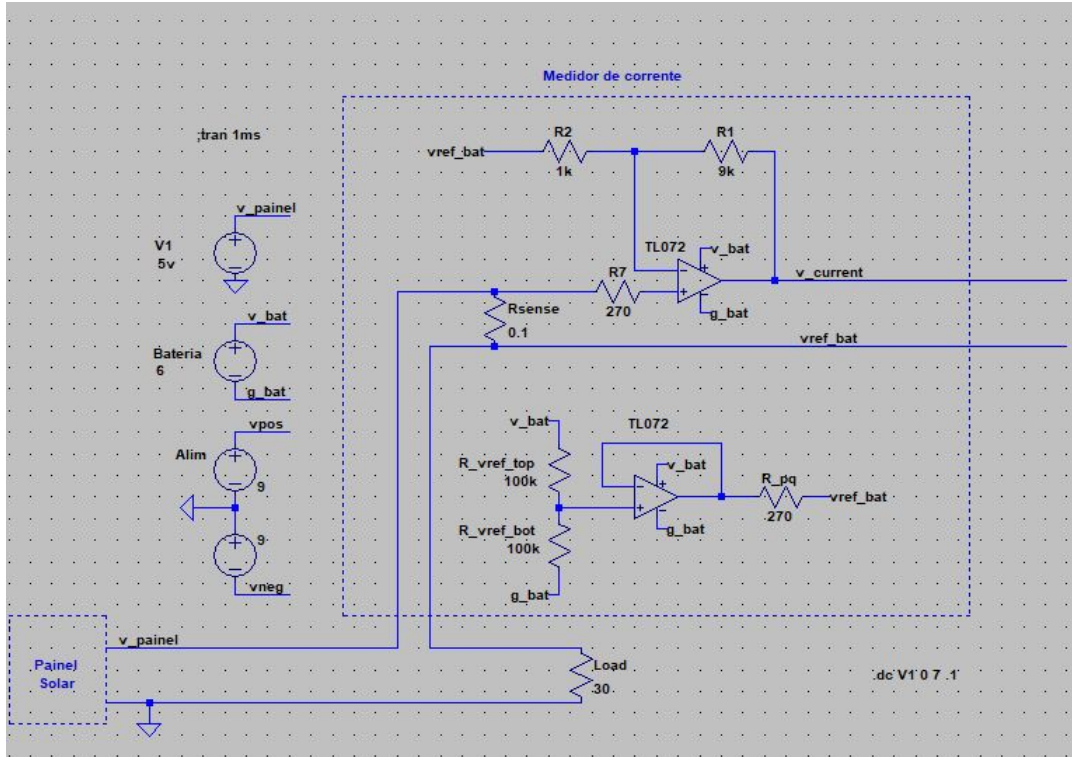


Figura 38 – Circuito Sensor de Corrente Implementado

O circuito de corrente será o responsável por enviar dados ao microcontrolador que processará a informação para que a onda *PWM* possa ter seu ciclo de trabalho alterado de forma a maximizar a potência extraída pelo painel. Para que o microcontrolador possa realizar a leitura correta da variação de tensão, faz-se necessário ajustar as referências de ambos os circuitos e para isso um circuito subtrator (Figura 39) foi montado para ajustar a diferença entre as tensões  $V\_current$  e  $vref\_bat$  em relação ao *ground* geral do circuito do painel solar/microcontrolador.

Simulando o conjunto sensor de corrente e o circuito mudança de referência, obteve-se a Figura 40 que mostra o comportamento da tensão de saída do circuito, referenciado ao *ground* geral do sistema. O sensor de corrente apresenta uma inclinação um pouco distinta da real curva de corrente, porém tal imprecisão não se mostra tão grave para a aplicação a ser realizada.



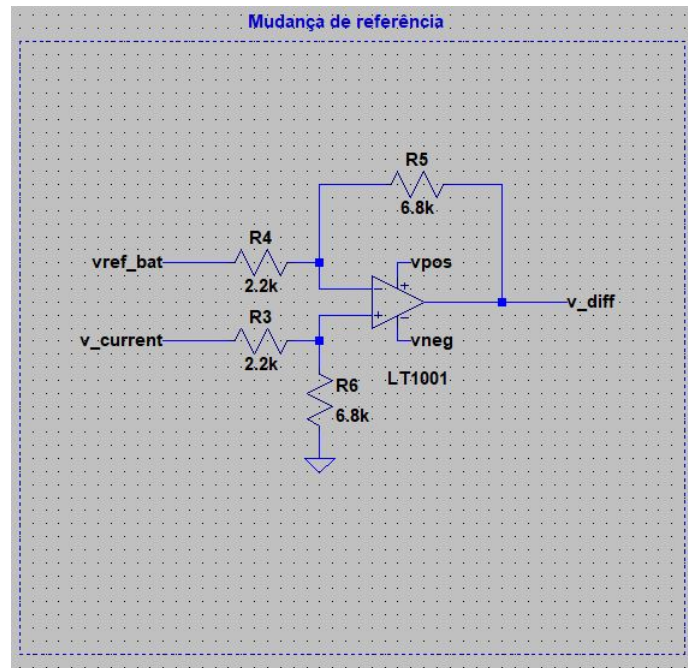


Figura 39 – Circuito Mudança de Referência

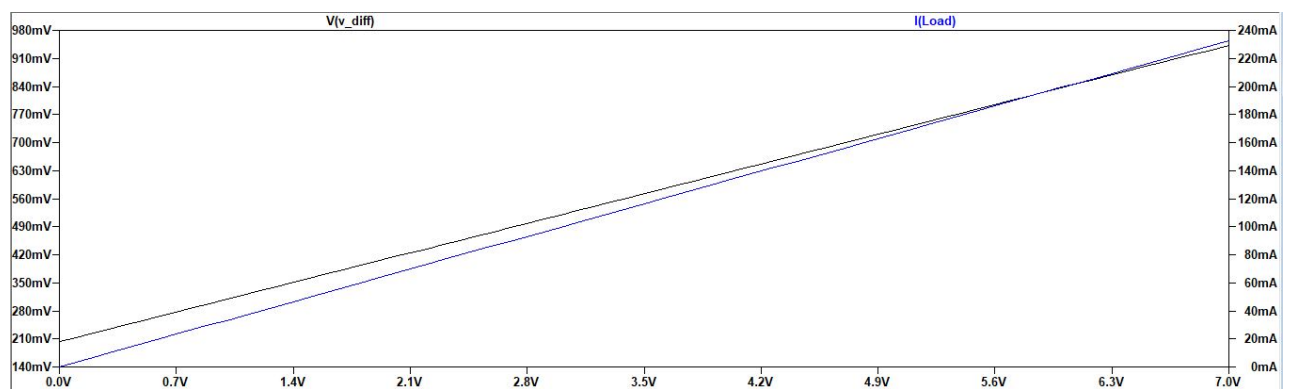


Figura 40 – Simulação Tensão Painei vs (Tensão na Saída do Sensor)/(Corrente em Rsens)

### 4.3.2 Montagem Sensor de Corrente

O sensor foi montado em conformidade com o processo metodológico descrito em Seção 3.1.3. Seguem as imagens obtidas via osciloscópio:



Figura 41 – Onda PWM com  $D=15.8\%$  e Tensão na Saída do Sensor de Corrente



Figura 42 – Onda PWM com  $D=22.4\%$  e Tensão na Saída do Sensor de Corrente



Figura 43 – Onda PWM com D=45.4% e Tensão na Saída do Sensor de Corrente

Tabela 8 – Relação de Tensão do Sensor com Corrente Extraída de Paine Solar

Corrente(mA)	Tensão do Sensor (mV)	Tensão do Paine (V)	duty cycle (%)
3,69	13,4	5,58	2,6
6,72	17,5	5,52	5,3
14,71	30,7	5,40	7,9
46,00	81,3	4,87	15,8
67,07	119	4,24	22,4
74,85	122	3,76	27,6
76,70	134	3,51	30,3
78,08	125	2,83	38,2
78,40	131	2,29	45,4
79,92	129	1,72	57,9
80,78	131	1,08	88,2

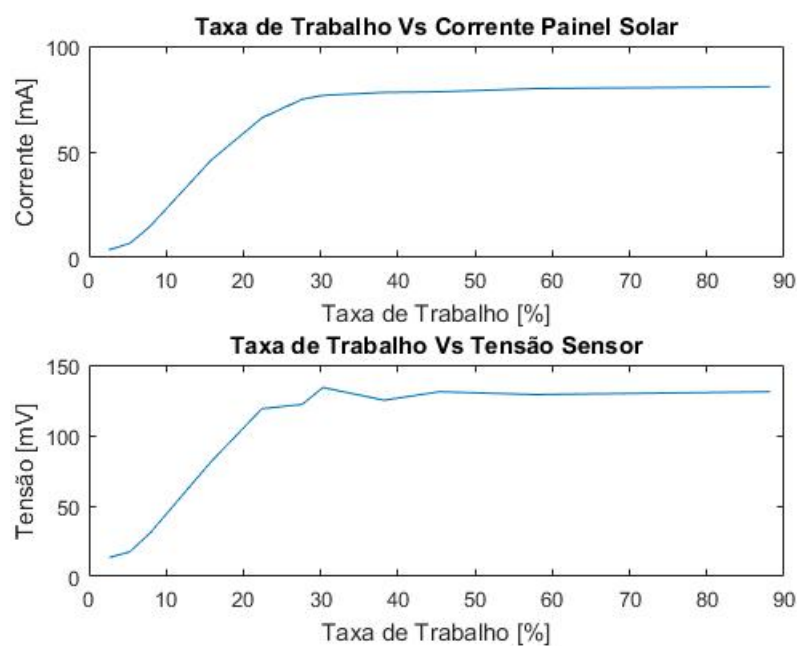


Figura 44 – Relação entre *duty cycle* vs (Tensão no sensor)/(Corrente na saída do paine)

Através das imagens obtidas via osciloscópio fica clara a presença de ruído. As imagens mostraram que o fato de o sensor de corrente estar presente na mesma referência do gerador de ondas PWM (microcontrolador), pode afetar a corrente lida pelo sensor. Com os valores médios de tensão do sensor e os valores de corrente medidos através de um amperímetro em série com o painel solar, obtemos a Tabela 8 e o gráfico da Figura 44. A Figura 44 mostra a relação entre corrente fluindo na saída do painel solar e a tensão proporcional a corrente na saída do sensor de corrente. A escala de corrente e de tensão não estão na mesma proporção devido a um ganho do amplificador operacional bem como a uma tensão de *offset* presente na saída do sensor de corrente. A relação entre a tensão na saída do sensor e a corrente na saída do painel está coerente com o esperado, apesar do ruído presente.

### 4.3.3 Sensor de Tensão

Para a análise da tensão na saída do painel solar, um divisor de tensão resistivo foi usado. O esquemático da Figura 45 representa o sensor de tensão para controle da potência extraída do painel. Dado que a tensão do painel flutua aproximadamente entre 0-7 Volts e o conversor analógico-digital do MSP430 converte tensões de até 3.3V, o divisor resistivo mostra-se suficiente para o controle.

$$V_{out} = \frac{R3}{R1 + R2} * V_{in} \quad (4.1)$$

$$V_{out} = \frac{V_{in}}{3}$$

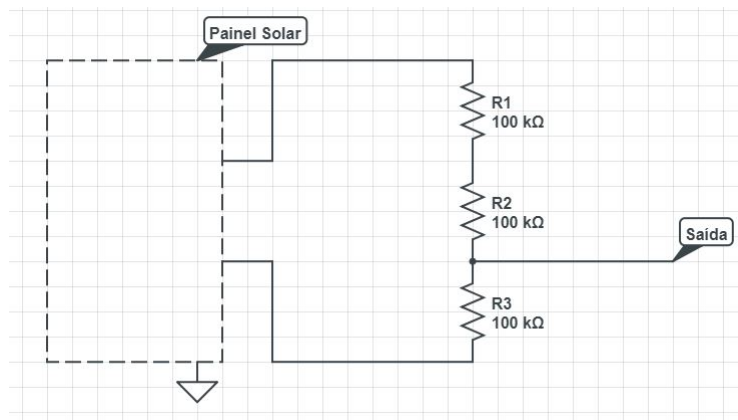


Figura 45 – Divisor Resistivo

## 4.4 Microcontrolador

O último passo para a conclusão do projeto dá-se na automatização dos algoritmos para maximização da potência do painel solar. Os algoritmos foram bem descritos na Seção 2.4. Para que ocorra o correto funcionamento do algoritmo, primeiramente é necessário que uma conversão analógica digital seja feita e o sinal provindo dos sensores de corrente e tensão sejam devidamente tratados e digitalmente analisados pelo microcontrolador. O compilador CCS foi utilizado para que a conversão e o algoritmo relativo ao MPPT fossem feitos.

### 4.4.1 Conversão Analógico Digital

A conversão analógica digital foi realizada a partir dos sinais emitidos pelos sensores de corrente e de tensão. O primeiro passo para análise dos dados foi obter a relação de conversão analógico/digital através de uma regressão linear de valores relativos a tensão, corrente e potência. Seguem abaixo os dados da regressão linear:

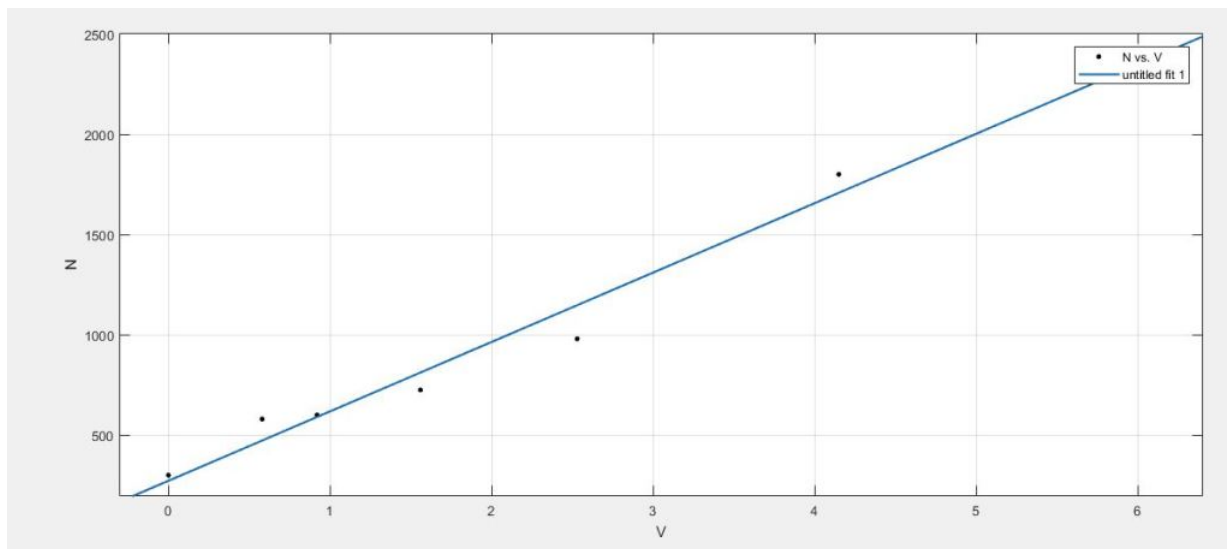


Figura 46 – Conversão Tensão [Volts] x Amostras Digitais [N]

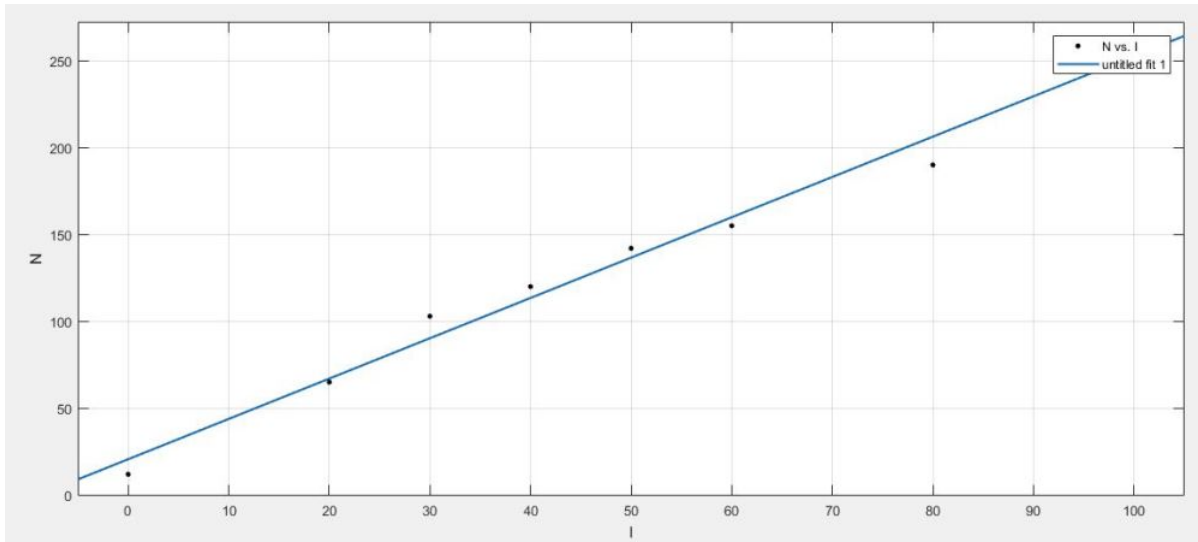


Figura 47 – Conversão Corrente [mA] x Amostras Digitais [N]

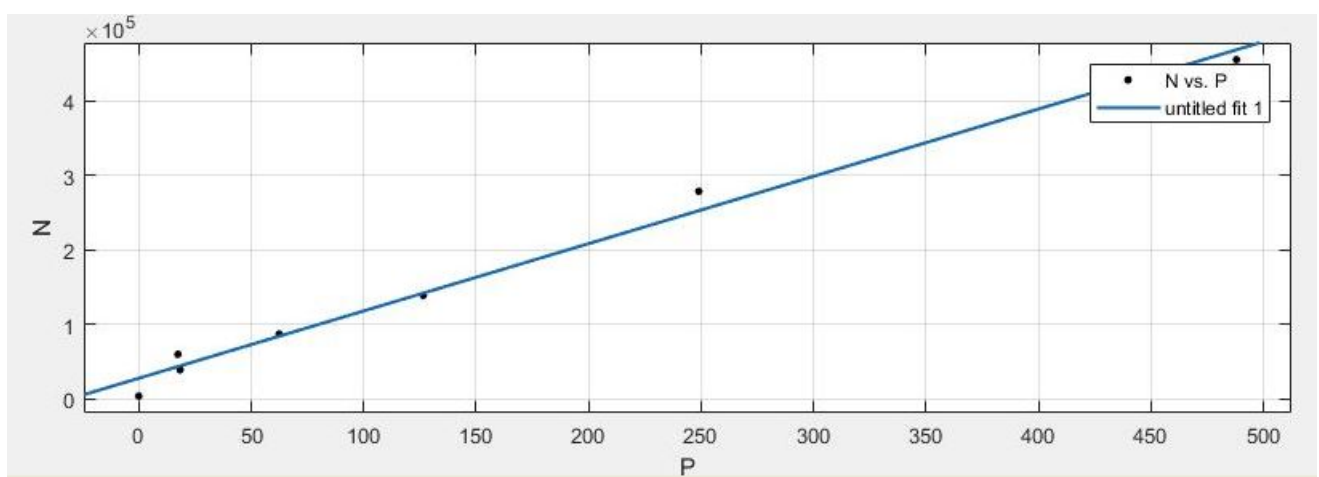


Figura 48 – Conversão Potência [mW] x Amostras Digitais [N]

$$N_v = 346 * V[volts] + 272,1 \quad (4.2)$$

$$N_i = 2,321 * I[mA] + 20,64 \quad (4.3)$$

$$N_p = 905,2 * P[mW] + 27570 \quad (4.4)$$

Abaixo observa-se os sinais de corrente e tensão já digitalizados. O sinal digital apresenta a mesma característica de ruído presente no sinal observado via osciloscópio (Figura 41). O código usado para realizar a conversão digital e o controle do *duty cycle* do microcontrolador via botões, está disposto no Apêndice A.2.

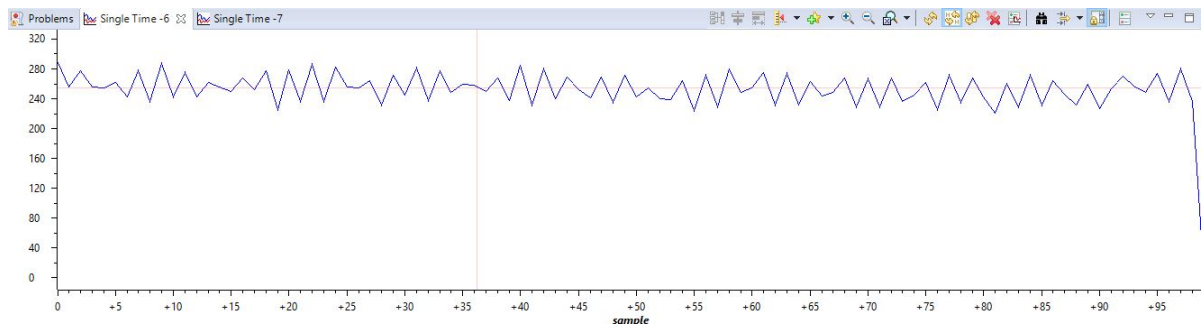


Figura 49 – Sinal Digitalizado Sensor de Corrente Funcionando com o Painel Solar ( $N_i$  x tempo)

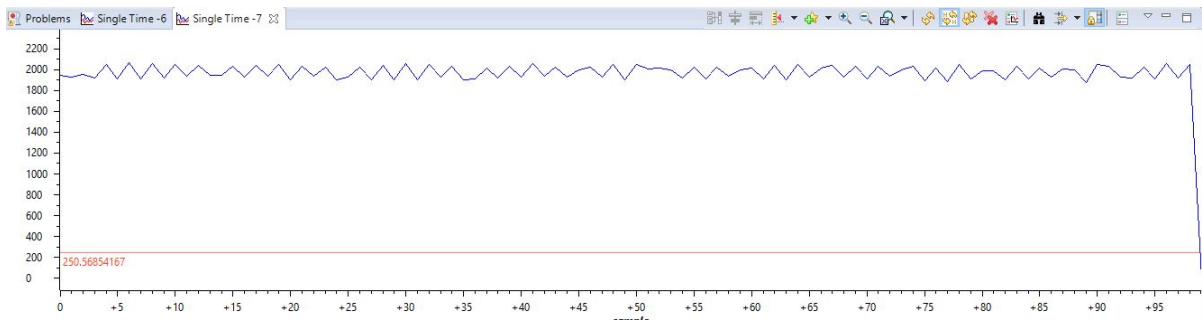


Figura 50 – Sinal Digitalizado Sensor de Tensão Funcionando com o Painel Solar ( $N_v$  x tempo)

#### 4.4.2 Algoritmo Perturba e Observa

O último passo para que o controlador de carga funcione, foi a implementação do algoritmo responsável por observar a relação corrente/tensão da entrada e analisar o valor do *duty cycle* que maximiza a potência retirada do painel. O código relativo ao algoritmo segue disposto em Apêndice A.3.

As figuras abaixo mostram o algoritmo P&O em seu pleno funcionamento. A Figura 51 exibe o comportamento do sinal de tensão na entrada do painel solar enquanto o controlador altera o *duty cycle* procurando a máxima eficiência. O sinal de tensão varia fortemente e tem dificuldades em encontrar um valor ótimo. A Figura 52 revela o sinal de corrente variando, também, de forma extremamente instável. A Figura 53, referente a potência do sistema, também varia de forma abrupta e em torno da máxima potência. As figuras 54, 55 e 56 demonstram o comportamento de tensão, corrente e potência quando variada a luminosidade ambiente.

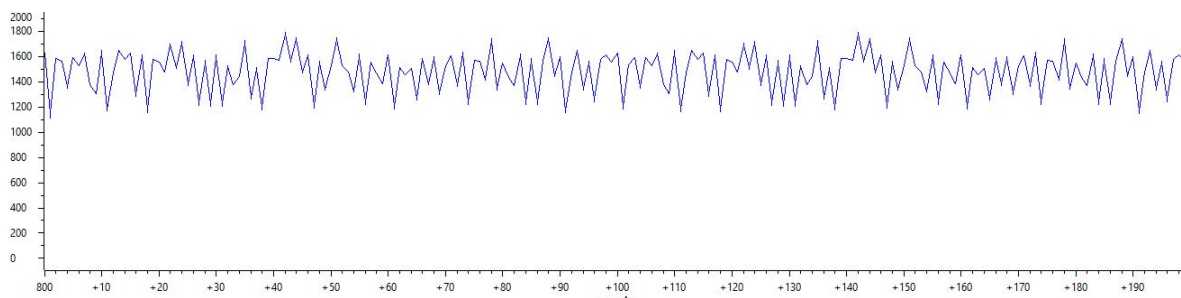


Figura 51 – Relação do Sinal de Tensão Variando de Acordo com Algoritmo P&O com luz contante (Nv x tempo)

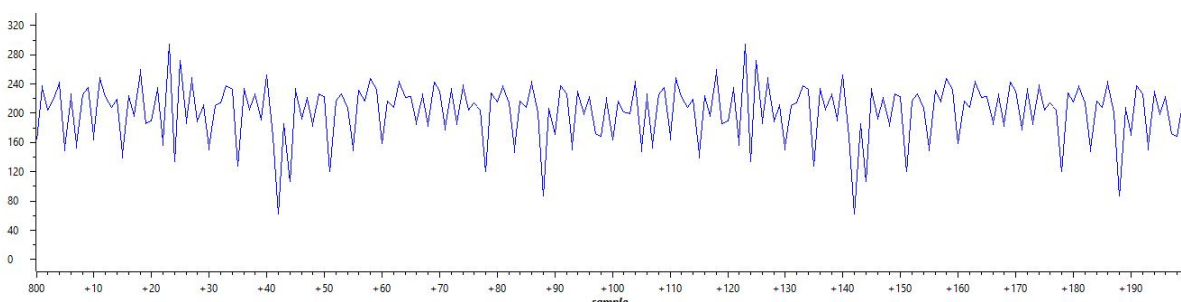


Figura 52 – Relação do Sinal de Corrente Variando de Acordo com Algoritmo P&O com luz constante (Ni x tempo)



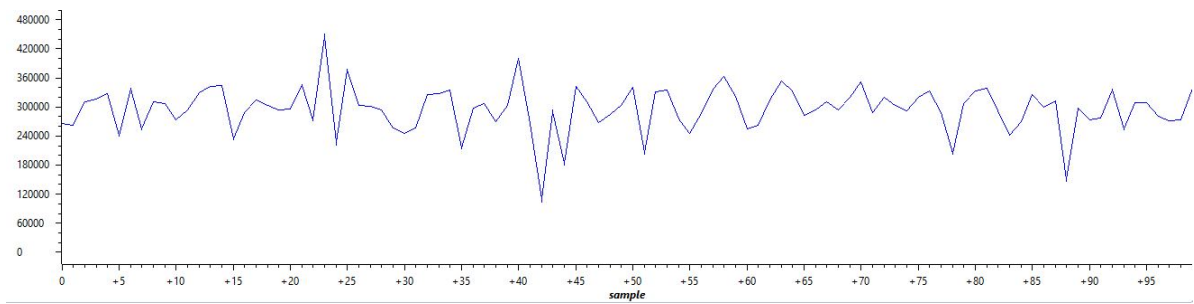


Figura 53 – Relação do Sinal de Potência Variando de Acordo com Algoritmo P&O com luz constante ( $N_p \times \text{tempo}$ )

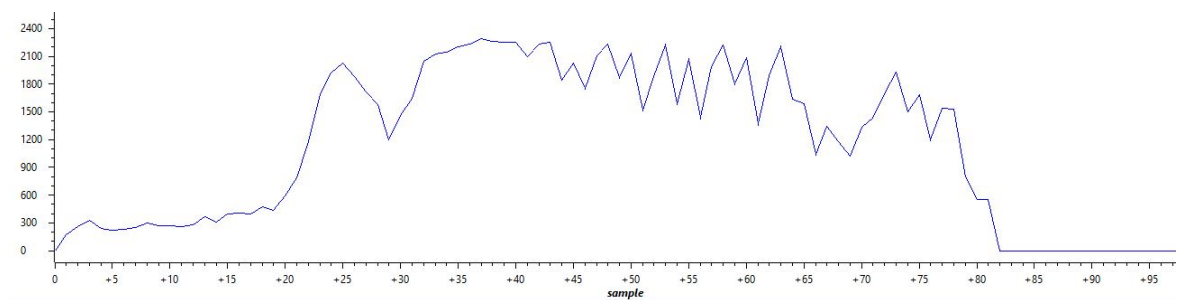


Figura 54 – Relação do Sinal de Tensão Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa ( $N_v \times \text{tempo}$ )

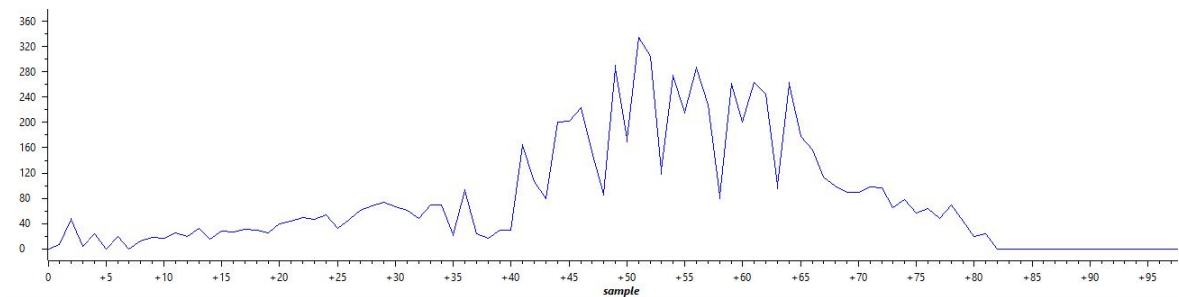


Figura 55 – Relação do Sinal de Corrente Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa ( $N_i \times \text{tempo}$ )

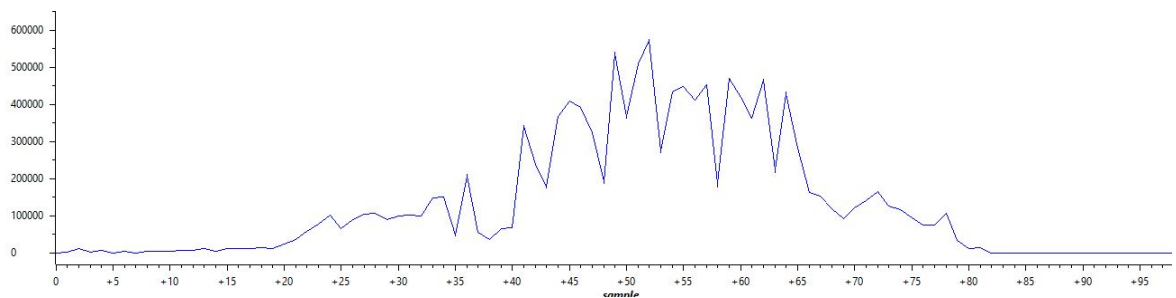


Figura 56 – Relação do Sinal de Potência Variando de Acordo com Algoritmo P&O com alteração da intensidade luminosa ( $N_p \times \text{tempo}$ )

Observamos nas imagens sem variação da luminosidade que o valor médio de tensão discretizada segue em torno de  $N_v = 1500$  o que dá em cerca de 3,548V de acordo com Equação 4.2. A corrente média gira em torno de  $N_i = 180$  o que dá em cerca de 68,66 mA (Equação 4.3). A potência média retirada do painel é aproximadamente  $N_p = 280000$  equivalente a 278,85 mW (Equação 4.4).

Evidencia-se com os dados retirados do microcontrolador que o algoritmo executado tende a levar o sistema para o ponto de máxima potência, porém de forma ruidosa. O ruído provocado pelo sensor de corrente e por outros elementos intrínsecos ao circuito provocam uma perturbação no fluxo natural do algoritmo.

## 5 Conclusão

O trabalho de conclusão de curso aqui apresentado teve como motivação o estudo de controladores universais de carga para dessa forma levar energia elétrica a zonas periféricas do globo terrestre de forma barata e eficiente. A implementação de um controlador de carga via MPPT foi o objetivo principal ao longo do trabalho.

Para compreender o mecanismo de controle de carga em painéis fotovoltaicos, foi proposta a implementação de um controlador funcionando através do algoritmo de rastreamento do ponto de máxima transferência de potência MPPT. O projeto foi dividido em pequenas partes para um melhor controle e efetividade das montagens. As principais partes do projeto foram o desenvolvimento do conversor DC/DC elevador e abaixador de tensão SEPIC, sensores de corrente e tensão e o controle do conversor DC /DC usando o microcontrolador MSP430. Ao longo do trabalho de desenvolvimento, o ponto mais complexo para o funcionamento do protótipo foi a implementação dos diversos sensores requeridos para leitura de dados. O sensor de corrente mostrou-se ruidoso e para um funcionamento adequado faz-se necessário a aplicação de filtro para uma maior precisão. Observou-se o funcionamento do algoritmo perturba e observa controlando a potência extraída do painel solar e apesar dos ruídos existentes há uma tendência do controlador ao ponto de máximo.

Uma continuação lógica deste trabalho dá-se no estudo das diversas fontes de ruído presentes no projeto, bem como ao estudo de melhores sensores e algoritmos para que o funcionamento do controlador de carga via MPPT atue de forma mais robusta frente as diversas perturbações. Após devida compreensão das fontes de problemas descritos no trabalho, o estudo de um controlador de carga universal que possua a flexibilidade requerida para projetos como *Smat Village*, proposto pela IEEE e devidamente descritos em Seção 1.1, seria o ideal para um grande avanço na atual conjuntura global do tema apresentado.



# Referências

B.V, V. E. Which solar charge controller: Pwm or mppt? In: . De Paal 35,1351 JG Almere, The Netherlands: [s.n.], 2014. p. 1–14. Citado 2 vezes nas páginas 9 e 24.

G.WALKER. Evaluating mppt converter topologies using a matlab pv model. Dept of Computer Science and Electrical Engineering, University of Queensland, Australia., 2012. Nenhuma citação no texto.

HARTS, D. *Power Electronics: Projects analysis*. [S.l.]: McGraw-Hill, 2011. Citado 8 vezes nas páginas 9, 25, 27, 28, 29, 30, 32 e 34.

IEA. 2017. Accessed: 23-10-2017. Disponível em: <<http://www.iea.org/energyaccess/>>. Citado na página 15.

JAGER, K. et al. Solar energy fundamentals, technology, and systems. In: . [S.l.: s.n.], 2014. p. 378. Nenhuma citação no texto.

KOTAK, V.; TYAGI, P. Dc to dc converter in maximum power point tracker. University of Technology Thanyaburi, v. 2, 2013. ISSN 2320-3765. Nenhuma citação no texto.

SAHA, S. S. A universal charge controller for integrating distributed energy resources. Seattle, WA, USA, p. 7, 2017. Disponível em: <<http://ieeexplore.ieee.org/document/7857320/>>. Citado 4 vezes nas páginas 9, 15, 16 e 17.

SERA, D.; TEODORESCU, R.; RODRIGUES, P. Pv panel model based on datasheet values. Aalborg University, Institute of Energy Technology, DK-9220 Aalborg, Denmark., 2007. Citado 2 vezes nas páginas 21 e 22.

TI, T. I. *Users Guide MSP43x5xx Family*. Texas Instruments, Post Office Box 655303, Dallas, Texas 75265, 2008. Citado 3 vezes nas páginas 9, 33 e 36.

UCURRENT Gold. Disponível em: <<https://www.eevblog.com/projects/ucurrent/>>. Citado na página 52.

YADAV<sup>1</sup>, A. K.; S. THIRUMALIAH<sup>2</sup>; G. HARITHA<sup>3</sup>. Comparison of mppt algorithms for dc-dc converters based pv systems. PG Scholar, St. Johns College of Engineering and Technology, 2012. ISSN 2278 – 8875. Citado 3 vezes nas páginas 9, 37 e 38.



# APÊNDICE A – Apêndice

## A.1 Código Variação da Taxa de Trabalho via Botões: MSP430

```

1 #include <msp430.h>
2 #include "lib/clock.h"
3 #include "lib/pmm.h"
4 #include "lib/port.h"
5
6 #define LED2          BIT7
7 #define LED2_OUT      P4OUT
8 #define LED1          BIT0
9 #define LED1_OUT      P1OUT
10
11
12 #define    Num_of_Results    16
13
14 volatile unsigned int A0results[Num_of_Results];
15 volatile unsigned int A1results[Num_of_Results];
16 volatile unsigned int A2results[Num_of_Results];
17
18 volatile signed int Vf;
19 volatile signed int I;
20 volatile signed int Vi;
21 volatile signed int Ii;
22 volatile signed int P;
23 volatile signed int Pi;
24 volatile signed int dp;
25
26 void delay2 () {
27     int a=0xFFFF;
28     while(a)
29         a--;
30 }
31
32 float a;
33
34 int main(void)
35 {
36     // setar outputs
37     P4DIR |= LED2;                                // Set P1.6 (LED) to
38     output direction
39     LED2_OUT &= ~LED2;

```

```

40     P1DIR |= LED1;
41     LED1_OUT &= ~LED1;
42
43
44
45     int adc;
46     WDTCIL = WDIPW + WDIHOLD;           // Stop WDT
47
48     SetVCoreUp(1);
49     SetVCoreUp(2);
50     SetVCoreUp(3);
51
52     clockInit();                         // Initialize clock system
53     // clockSetDCO(5MHz);                // DCO runs @25MHz
54     clockSelect(DCO, MCLK);              // Route DCO to MCLK
55     clockSelect(XT1, ACLK);              // Set ACLK to XT1 (32768Hz)
56     clockSelect(XT2, SMCLK);             // Set SMCLK to XT2 (4MHz)
57
58     portInit();
59
60
61
62
63     P1IE |= BIT1;                        // P1.3 interrupt enabled
64     P1IES |= BIT1;                       // P1.3 Hi/lo edge
65     P1IFG &= ~BIT1;                     // P1.3 IFG cleared
66
67     P2IE |= BIT1;                        // P1.3 interrupt enabled
68     P2IES |= BIT1;                       // P1.3 Hi/lo edge
69     P2IFG &= ~BIT1;                     // P1.3 IFG cleared
70
71     __enable_interrupt();
72
73
74
75     P1DIR |= BIT2;                       // P1.2
76     P1SEL |= BIT2;                       // P1.2
77     TA0CCR0 = 40-1;                      // PWM Period
78     TA0CTL1 = OUTMOD_7;                  // CCR1 reset/set
79     TA0CCR1 = 1;                         // CCR1 PWM duty cycle
80     TA0CTL = TASSEL_2 + MC_1 + TACLK;    // SMCLK, up mode, clear TAR
81 }
82 #pragma vector = PORT1_VECTOR
83 __interrupt void port1(void){
84     int p;
85     p = __even_in_range(P1IV,0x10);
86     switch(p){

```



```

87     case 0x0: break; //sem int
88     case 0x2: break; // 0
89     case 0x4:
90         if (TA0CCR1>39){
91             TA0CCR1=40;
92             break;
93         }
94         TA0CCR1++;
95
96
97
98     break; // 1
99     case 0x6: break; // 2
100    case 0x8: break; //3
101    case 0xA: break; //4
102    case 0xC: break; //5
103    case 0xE: break; //6
104    case 0x10: break; //7
105 }
106 }
107
108 #pragma vector = PORT2_VECTOR
109 __interrupt void port2(void){
110     int p;
111     p = __even_in_range(P2IV,0x10);
112     switch(p){
113         case 0x0: break;
114         case 0x2: break;
115         case 0x4:
116             if (TA0CCR1<1){
117                 TA0CCR1==0;
118                 break;
119             }
120             TA0CCR1--;
121
122
123             break;
124         case 0x6: break;
125         case 0x8: break;
126         case 0xA: break;
127         case 0xC: break;
128         case 0xE: break;
129         case 0x10: break;
130     }
131 }

```

## A.2 Código Conversão ADC via Botões: MSP430

```

1 #include <msp430.h>
2 #include "lib/clock.h"
3 #include "lib/pmm.h"
4 #include "lib/port.h"
5
6 #define LED2          BIT7
7 #define LED2_OUT      P4OUT
8 #define LED1          BIT0
9 #define LED1_OUT      P1OUT
10
11
12
13 static unsigned int b=0;
14
15
16
17 volatile unsigned int Vf[100];
18 volatile unsigned int I[100];
19
20
21 static unsigned int index = 0;
22
23
24 int main(void)
25 {
26     // setar outputs
27     P4DIR |= LED2;                      // Set P1.6 (LED) to output
28     LED2_OUT &= ~LED2;
29     direction
30     P1DIR |= LED1;
31     LED1_OUT &= ~LED1;
32
33
34
35
36     WDCTL = WDIPW + WDTHOLD;           // Stop WDT
37
38     SetVCoreUp(1);
39     SetVCoreUp(2);
40     SetVCoreUp(3);
41
42     clockInit();                        // Initialize clock system
43     // clockSetDCO(5MHz);                // DCO runs @25MHz
44     clockSelect(DCO, MCLK);             // Route DCO to MCLK
45     clockSelect(XT1, ACLK);             // Set ACLK to XT1 (32768Hz)

```

```

46     clockSelect(XT2, SMCLK);                // Set SMCLK to XT2 (4MHz)
47
48     portInit();
49
50
51
52
53     P1IE |= BIT1;                          // P1.3 interrupt enabled
54     P1IES |= BIT1;                        // P1.3 Hi/lo edge
55     P1IFG &= ~BIT1;                      // P1.3 IFG cleared
56
57     P2IE |= BIT1;                          // P1.3 interrupt enabled
58     P2IES |= BIT1;                        // P1.3 Hi/lo edge
59     P2IFG &= ~BIT1;                      // P1.3 IFG cleared
60
61     // __enable_interrupt();
62
63     ///////////////PWM timer A ///////////////////
64
65     P1DIR |= BIT2;                        // P1.2
66     P1SEL |= BIT2;                        // P1.2
67     TA0CCR0 = 40-1;                      // PWM Period
68     TA0CCTL1 = OUTMOD_7;                  // CCR1 reset/set
69     TA0CCR1 = 10;                        // CCR1 PWM duty cycle
70     TA0CTL = TASSEL_2 + MC_1 + TACLK;     // SMCLK, up mode, clear TAR
71
72     /////////////// Timer B Controle do fluxo de dados ADC ///////////////////
73
74     P1DIR |= 0x01;                        // Set P1.0 to output
75     direction
76     TBCCTL0 = CCIE;                      // TRCCR0 interrupt enabled
77     TBCCR0 = 5000-1;                     //Toggle rate = 32768Hz
78     //(2*16384) = 1Hz
79
80     // ACLK = TBCLK = 32kHz, MCLK = SMCLK = default
81     DCO ~1.045MHz
82
83     TBCTL = TBSSEL_1 + MC_1 + TBCLK;     // ACLK, upmode, clear TBR
84
85
86     /////////////// Conversor ADC ///////////////////
87
88     P6SEL = 0x07;                        // Enable A/D channel inputs
89     ADC12CTL0 = ADC12ON+ADC12MSC+ADC12SHT0_2; // Turn on ADC12, extend
90     sampling time
91     // to avoid overflow of results
92     ADC12CTL1 = ADC12SHP+ADC12CONSEQ_1;   // Use sampling timer, single
93     sequence

```

```

88     ADC12MCTL0 = ADC12INCH_1;                // ref+=AVcc, channel = A0
89     ADC12MCTL1 = ADC12INCH_2+ADC12EOS;        // ref+=AVcc,
channel = A1
90     //ADC12MCTL2 = ADC12INCH_2;              // ref+=AVcc, channel = A2
91     ADC12IE = ADC12IE1;                      // Enable ADC12IFG.2
92     ADC12CTL0 |= ADC12ENC;                   // Enable conversions
93
94
95
96     __bis_SR_register(GIE);                  // Enter LPM0, Enable interrupts
97     //__no_operation();                      // For debugger
98
99
100    volatile signed int di;
101    volatile signed int dv;
102    volatile unsigned int Vi=0;
103    volatile unsigned int ii=0;
104    int voltage=0;
105    int current=0;
106    float M=0.0;
107
108    static unsigned int count;
109    while(1){
110    }
111
112
113
114
115 }
116
117
118 #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
119 #pragma vector=ADC12_VECTOR
120 __interrupt void ADC12ISR (void)
121 #elif defined(__GNUC__)
122 void __attribute__((interrupt(ADC12_VECTOR))) ADC12ISR (void)
123 #else
124 #error Compiler not supported!
125 #endif
126 {
127
128
129     switch(__even_in_range(ADC12IV,34))
130     {
131     case 0: break;                // Vector 0: No interrupt
132     case 2: break;                // Vector 2: ADC overflow
133     case 4: break;                // Vector 4: ADC timing

```

```

overflow
134     case 6: break;                                // Vector 6:  ADC12IFG0
135
136     case 8:                                        // Vector 10:  ADC12IFG2
137
138
139         I[index] = ADC12MEM0;                        // Move A1 results , IFG is cleared
140         Vf[index] = ADC12MEM1;                      // Move A2 results , IFG is cleared
141         // A3results[index] = ADC12MEM3;            // Move A3 results ,
IFG is cleared
142         index++;                                    // Increment results index ,
modulo; Set Breakpoint1 here
143         b=1;
144         if (index == 99)
145         {
146
147             (index = 0);
148         }
149     case 12: break;                                // Vector 12:
ADC12IFG3
150     case 14: break;                                // Vector 14:  ADC12IFG4
151     case 16: break;                                // Vector 16:  ADC12IFG5
152     case 18: break;                                // Vector 18:  ADC12IFG6
153     case 20: break;                                // Vector 20:  ADC12IFG7
154     case 22: break;                                // Vector 22:  ADC12IFG8
155     case 24: break;                                // Vector 24:  ADC12IFG9
156     case 26: break;                                // Vector 26:  ADC12IFG10
157     case 28: break;                                // Vector 28:  ADC12IFG11
158     case 30: break;                                // Vector 30:  ADC12IFG12
159     case 32: break;                                // Vector 32:  ADC12IFG13
160     case 34: break;                                // Vector 34:  ADC12IFG14
161     default: break;
162 }
163 }
164
165 // Timer B0 interrupt service routine
166 #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
167 #pragma vector=TIMERB0_VECTOR
168 __interrupt void TIMERB1_ISR(void)
169 #elif defined(__GNUC__)
170 void __attribute__((interrupt(TIMERB0_VECTOR))) TIMERB1_ISR (void)
171 #else
172 #error Compiler not supported!
173 #endif
174 {
175     P1OUT ^= 0x01;                                // Sinalizador de Frequencia de
amostragem

```

```

176     ADC12CTL0 |= ADC12SC;
177
178 }
179
180 #pragma vector = PORT1_VECTOR
181 __interrupt void port1(void){
182     int p;
183     p = __even_in_range(P1IV,0x10);
184     switch(p){
185         case 0x0: break; //sem int
186         case 0x2: break; // 0
187         case 0x4:
188             if (TA0CCR1>39){
189                 TA0CCR1=40;
190                 break;
191             }
192             TA0CCR1++;
193
194
195
196             break; // 1
197         case 0x6: break; // 2
198         case 0x8: break; //3
199         case 0xA: break; //4
200         case 0xC: break; //5
201         case 0xE: break; //6
202         case 0x10: break; //7
203     }
204 }
205
206 #pragma vector = PORT2_VECTOR
207 __interrupt void port2(void){
208     int p;
209     p = __even_in_range(P2IV,0x10);
210     switch(p){
211         case 0x0: break;
212         case 0x2: break;
213         case 0x4:
214             if (TA0CCR1<1){
215                 TA0CCR1==0;
216                 break;
217             }
218             TA0CCR1--;
219
220
221             break;
222         case 0x6: break;

```

```
223     case 0x8: break;
224     case 0xA: break;
225     case 0xC: break;
226     case 0xE: break;
227     case 0x10: break;
228 }
229 }
```

## A.3 Código Algoritmo Perturba e Observa

```
1
2 #include <msp430.h>
3 #include "lib/clock.h"
4 #include "lib/pmm.h"
5 #include "lib/port.h"
6
7 #define LED2          BIT7
8 #define LED2_OUT      P4OUT
9 #define LED1          BIT0
10 #define LED1_OUT      P1OUT
11
12
13
14 static unsigned int b=0;
15
16
17
18 volatile unsigned long Vf;
19 volatile unsigned long I;
20
21
22 static unsigned int index = 0;
23
24
25 int main(void)
26 {
27     // setar outputs
28     P4DIR |= LED2;                               // Set P1.6 (LED) to output
29     direction
30     LED2_OUT &= ~LED2;
31
32     P1DIR |= LED1;
33     LED1_OUT &= ~LED1;
34
35
36
```

```

37  WDCTL = WDIPW + WDTHOLD;           // Stop WDT
38
39  SetVCoreUp(1);
40  SetVCoreUp(2);
41  SetVCoreUp(3);
42
43  clockInit();                        // Initialize clock system
44  // clockSetDCO(5MHz);                // DCO runs @25MHz
45  clockSelect(DCO, MCLK);              // Route DCO to MCLK
46  clockSelect(XT1, ACLK);              // Set ACLK to XT1 (32768Hz)
47  clockSelect(XT2, SMCLK);            // Set SMCLK to XT2 (4MHz)
48
49  portInit();
50
51
52
53
54  P1IE |= BIT1;                       // P1.3 interrupt enabled
55  P1IES |= BIT1;                      // P1.3 Hi/lo edge
56  P1IFG &= ~BIT1;                     // P1.3 IFG cleared
57
58  P2IE |= BIT1;                       // P1.3 interrupt enabled
59  P2IES |= BIT1;                      // P1.3 Hi/lo edge
60  P2IFG &= ~BIT1;                     // P1.3 IFG cleared
61
62  // __enable_interrupt();
63
64  ///////////////////////////////////////////////////PWM timer A ///////////////////////////////////
65
66  P1DIR |= BIT2;                      // P1.2
67  P1SEL |= BIT2;                      // P1.2
68  TA0CCR0 = 40-1;                     // PWM Period
69  TA0CTL1 = OUTMOD_7;                 // CCR1 reset/set
70  TA0CCR1 = 10;                       // CCR1 PWM duty cycle
71  TA0CTL = TASSEL_2 + MC_1 + TACLK;   // SMCLK, up mode, clear TAR
72
73  /////////////////////////////////////////////////// Timer B Controle do fluxo de dados ADC ///////////////////////////////////
74
75  P1DIR |= 0x01;                      // Set P1.0 to output
76  direction
77  TBCCTL0 = CCIE;                     // TRCCR0 interrupt enabled
78  TBCCR0 = 16384-1;                  // Toggle rate = 32768Hz
79  //(2*16384) = 1Hz
80  // ACLK = TBCLK = 32kHz, MCLK = SMCLK = default
81  DCO ~1.045MHz
82  TBCTL = TBSSEL_1 + MC_1 + TBCLR;    // ACLK, upmode, clear TBR

```



```

81
82
83 /////////////// Conversor ADC ///////////////////
84
85 P6SEL = 0x07; // Enable A/D channel inputs
86 ADC12CTL0 = ADC12ON+ADC12MSC+ADC12SHT0_2; // Turn on ADC12, extend
sampling time
87 // to avoid overflow of results
88 ADC12CTL1 = ADC12SHP+ADC12CONSEQ_1; // Use sampling timer, single
sequence
89 ADC12MCTL0 = ADC12INCH_1; // ref+=AVcc, channel = A0
90 ADC12MCTL1 = ADC12INCH_2+ADC12EOS; // ref+=AVcc,
channel = A1
91 //ADC12MCTL2 = ADC12INCH_2; // ref+=AVcc, channel = A2
92 ADC12IE = ADC12IE1; // Enable ADC12IFG.2
93 ADC12CTL0 |= ADC12ENC; // Enable conversions
94
95
96
97 __bis_SR_register(GIE); // Enter LPM0, Enable interrupts
98 //__no_operation(); // For debugger
99 volatile unsigned long Pi;
100 volatile unsigned long P;
101 volatile signed long dp;
102 volatile unsigned long Vi=0;
103 volatile unsigned long ii=0;
104 volatile unsigned long Power[100]=0;
105 volatile unsigned long Voltage[100]=0;
106 volatile unsigned long Current[100]=0;
107
108
109 static unsigned int count;
110 while(1){
111
112     b=0; // Start convn - software trigger
113
114     P=(Vf*I);
115     Pi=Vi*ii;
116     dp=P-Pi;
117
118     if(dp>0){
119         if(Vf<Vi){
120             TA0CCR1++;
121             if(TA0CCR1>10)
122                 TA0CCR1=10;}
123         if(Vf>Vi){
124             TA0CCR1--;

```

```

125         if (TA0CCR1<2)
126             TA0CCR1=2;
127     }
128 }
129
130 if (dp<0){
131     if (Vf>Vi){
132         TA0CCR1++;
133         if (TA0CCR1>10)
134             TA0CCR1=10;}
135     if (Vf<Vi){
136         TA0CCR1--;
137         if (TA0CCR1<1)
138             TA0CCR1=1;}
139 }
140 ///// graficos ///
141     Power [count]=P;
142     Current [count]=I;
143     Voltage [count]=Vf;
144     count++;
145     if (count >99)
146         count=0;
147     ////////////////
148     Vi=Vf;
149     ii=I;
150
151
152     while (b==0);
153 }
154
155
156
157
158 }
159
160
161 #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
162 #pragma vector=ADC12_VECTOR
163 __interrupt void ADC12ISR (void)
164 #elif defined(__GNUC__)
165 void __attribute__ ((interrupt(ADC12_VECTOR))) ADC12ISR (void)
166 #else
167 #error Compiler not supported!
168 #endif
169 {
170
171

```

```

172     switch ( __even_in_range(ADC12IV,34) )
173     {
174         case 0: break;           // Vector 0: No interrupt
175         case 2: break;           // Vector 2: ADC overflow
176         case 4: break;           // Vector 4: ADC timing
177         case 6: break;           // Vector 6: ADC12IFG0
178
179         case 8: {                 // Vector 10: ADC12IFG2
180
181
182             I = ADC12MEM0;        // Move A1 results , IFG is cleared
183             Vf = ADC12MEM1;       // Move A2 results , IFG is cleared
184
185
186             // A3results[index] = ADC12MEM3;           // Move A3 results ,
187             IFG is cleared
188             b=1;
189             }
190         case 12: break;           // Vector 12:
191         case 14: break;           // Vector 14: ADC12IFG4
192         case 16: break;           // Vector 16: ADC12IFG5
193         case 18: break;           // Vector 18: ADC12IFG6
194         case 20: break;           // Vector 20: ADC12IFG7
195         case 22: break;           // Vector 22: ADC12IFG8
196         case 24: break;           // Vector 24: ADC12IFG9
197         case 26: break;           // Vector 26: ADC12IFG10
198         case 28: break;           // Vector 28: ADC12IFG11
199         case 30: break;           // Vector 30: ADC12IFG12
200         case 32: break;           // Vector 32: ADC12IFG13
201         case 34: break;           // Vector 34: ADC12IFG14
202         default: break;
203     }
204
205 // Timer B0 interrupt service routine
206 #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
207 #pragma vector=TIMERB0_VECTOR
208 __interrupt void TIMERB1_ISR(void)
209 #elif defined(__GNUC__)
210 void __attribute__((interrupt(TIMERB0_VECTOR))) TIMERB1_ISR (void)
211 #else
212 #error Compiler not supported!
213 #endif
214 {
215     P1OUT ^= 0x01;               // Sinalizador de Frequencia de

```

```
216     amostragem  
217     ADC12CTL0 |= ADC12SC;  
218 }
```